

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

FACULTAD DE INGENIERÍA

ESCUELA DE SISTEMAS



TEMA:

DESARROLLO E IMPLEMENTACIÓN DE APLICATIVO WEB PARA RESERVAS
DE PARQUEADEROS DE LA PONTIFICIA UNIVERSIDAD CATÓLICA DEL
ECUADOR.

ANGEL BERNARDO RUIZ ANDRADE

ALEGRÍA ROSALES TERÁN

DIRECTOR:

ING. DAMIÁN NICOLALDE

QUITO, NOVIEMBRE DEL 2016

Dedicatoria Alegría Rosales

Dedico este plan de disertación a mi abuelito Gustavo, porque siempre me ha dedicado todo su cariño y apoyo y siempre sonrío cada vez que lo veo, levantándome siempre el ánimo en cualquier momento. Por ser un modelo de trabajo duro y amor por las cosas que importan de verdad.

Además quiero dedicar este plan a mis abuelos Aída, Beatriz y Alberto que me cuidan desde el cielo y me guían para escoger siempre el mejor camino.

Agradecimiento Alegría Rosales

A mis padres que me han apoyado desde el inicio de mi carrera y por enseñarme que con esfuerzo se consigue el éxito. A mis hermanos que me han dado todo su cariño y siempre hacen que me exija ser una mejor persona.

A mi prima y mejor amiga Meli por siempre apoyarme cuando más lo necesito.

A mi director de tesis Damián Nicolalde quien siempre nos brindó ayuda y consejo facilitándonos con el soporte que necesitábamos. A nuestros lectores Andrés Jiménez y Oswaldo Espinosa por su colaboración y tiempo para ayudarnos con este proyecto.

A mis profesores que han impartido su conocimiento durante mi etapa universitaria y me han enseñado a pensar siempre en el servicio a la comunidad.

A mis amigos, por brindarme momentos inolvidables y por ayudarme siempre que los he necesitado. A mis compañeros de clase por compartir gratos momentos conmigo dentro y fuera de las aulas.

A Dios por guiarme durante esta trayectoria y por darme la fuerza necesaria para cumplir con este reto.

Finalmente, pero no menos importante, quiero agradecer a mi compañero de tesis Angel, ya que sin su colaboración, conocimiento y perseverancia, este trabajo no hubiese culminado con los mismos resultados.

Dedicatoria Angel Ruiz

Esta tesis la dedico en primer lugar a Dios quien ha sido mi guía para tomar un camino correcto en mi vida. A mi familia que me brindo las fuerzas necesarias para nunca dejarme caer en este, uno de mis sueños y por la comprensión, amor y cariño que me han brindado incondicionalmente a lo largo de mi carrera universitaria.

A mis padres Angel y Graciela, por ser mis pilares, un ejemplo de vida, inspiración y motivación constante día a día. A mi padre por ser mi compañía y creer en mí para salir siempre adelante. A mi madre, por demostrarme el verdadero amor y formar mi carácter con valores que me permitieron conseguir mis ideales y metas de vida.

A mis hermanos, Nikola y Mateo, por dedicarle a mi sueño tiempo y darme la oportunidad de ser un ejemplo para ellos, ser parte de cada momento en mi vida y permanecer incondicionales con paciencia, tocando mi corazón dispuestos a colaborar en todo momento.

A mi compañera y amiga Alegría Rosales, que en base a su apoyo supo contagiarme de entusiasmo y perseverancia a lo largo de nuestra carrera universitaria, extendiéndome su mano en cada momento sin dejar de lado nuestros objetivos, permitiéndome tener un apoyo constante dispuesto a contagiar alegría y lucha por conseguir todos nuestros propósitos.

Agradecimiento Angel Ruiz

Agradezco a Dios por brindarme la oportunidad de terminar mi carrera universitaria a manera de un triunfo personal y brindarme una familia perfecta a mi lado.

A mis padres por darme la fortaleza y apoyo necesario para seguir y no desfallecer en momento difíciles, siendo mi ejemplo de perseverancia.

A mis hermanos por ser mi verdadera muestra de amor y permitirme ser mi equilibrio de vida, infinitas gracias por velar por mi bienestar y motivarme diariamente a ser su ejemplo constante de vida.

A mis profesores, por brindarme conocimiento y enseñanza en todo momento, enriqueciéndome como profesional y como ser humano, motivándome a encontrar puntos clave en mi vida que promuevan mi formación y enaltezcan a nuestra querida Pontificia Universidad Católica del Ecuador.

A mi tutor, MSc. Damián Nicolalde, por su dedicación y firmeza, quien supo guiarme con conocimientos prácticos y teóricos, demostrándome que mis habilidades se transforman en fortalezas en base a esfuerzo y dedicación total, generando recompensas.

Finalmente deseo agradecer a todas las personas que colaboraron y contribuyeron a lograr esta meta final.

Con infinito agradecimiento y cariño

Angel Bernardo Ruiz Andrade

Resumen

En el siguiente documento se expone la disertación de grado que consiste en el desarrollo de un sistema en forma de aplicativo web para la reserva de parqueaderos desarrollada para la Pontificia Universidad Católica del Ecuador. El objetivo de este aplicativo es facilitar el proceso de parqueaderos dentro del campus de la universidad ofreciendo un sistema que permite la reserva de una plaza en un sector determinado a una hora determinada para las personas que tienen el beneficio de un parqueadero a su favor. Este sistema también facilita el control para los guardias de control y de recorrido para validar infracciones que se cometen en los parqueaderos.

1 CONTENIDOS

2	CAPÍTULO I: Introducción	9
2.1	Situación Actual.....	9
2.1.1	La Universidad	9
2.1.2	Análisis del proceso “ENTREGA DE ADHESIVOS DE PARQUEADERO”	11
2.2	Identificación del Problema	11
2.3	Alcance y Limitaciones	12
2.3.1	Alcance	12
2.3.2	Limitaciones	12
2.4	Análisis de Mercado	13
2.4.1	Encuesta.....	14
2.4.2	Tabulación y análisis	14
2.4.3	Técnicas de análisis	21
3	CAPÍTULO II: Marco Teórico.....	22
3.1	Herramientas de modelado	22
3.1.1	Metodología de desarrollo UWE.....	22
3.2	Herramientas de desarrollo de software.....	25
3.2.1	Herramientas de diseño	25
3.2.2	Herramientas de desarrollo.....	29
4	CAPÍTULO III: Diseño de la aplicación.....	37
4.1	Modelo de requerimientos	37
4.1.1	Definir actores	37
4.1.2	Casos de uso de la aplicación y sus relaciones.....	38
4.1.3	Actividades describiendo los casos de uso en detalle.....	40
4.2	Modelo de contenido	42
4.2.1	Diagrama de clases	42
4.3	Modelo de navegación	44
4.4	Modelo de Presentación.....	46
4.4.1	Pantalla de ingreso:.....	46
4.4.2	Pantalla principal	48

4.4.3	Creación de usuarios.....	49
4.4.4	Creación fechas de reservas.....	49
4.4.5	Reportes de infracciones.....	50
4.4.6	Reportes de reservas	50
4.4.7	Reporte de movimientos vehiculares.....	51
4.4.8	Editar información de usuario	51
4.4.9	Registro de ingresos y salidas.....	52
4.4.10	Informes de recorrido	52
4.4.11	Generar lista de vehículos por sector.....	53
4.4.12	Generar reserva por parte del usuario.....	53
4.4.13	Seleccionar plazas.....	54
4.4.14	Escoger horarios de plazas	54
4.4.15	Ver reservas realizadas	55
4.4.16	Ver reservas de usuarios	55
4.5	Modelo de Procesos	56
4.5.1	Ingreso al sistema	56
4.5.2	Creación de usuarios.....	56
4.5.3	Modificación de usuarios.....	57
4.5.4	Eliminar usuarios.....	57
4.5.5	Crear fechas de reservas	58
4.5.6	Eliminar fechas	58
4.5.7	Generar reportes	59
4.5.8	Registro de ingresos vehiculares	59
4.5.9	Registro de salidas vehiculares.....	60
4.5.10	Nuevos informes.....	60
4.5.11	Eliminar informes	61
4.5.12	Generar lista de vehículos por sector.....	61
4.5.13	Generar reservas	62
4.5.14	Anular reservas	62
4.5.15	Información reservas	63
4.5.16	Información infracciones	63
5	CAPÍTULO IV: Desarrollo e implementación del sistema web	64
5.1	CONFIGURACIÓN DE HERRAMIENTAS	64

5.1.1	Servidor de base de datos	64
5.1.2	Servidor web.....	64
5.1.3	phpMyAdmin	64
5.2	Herramientas del Sistema	65
5.2.1	Librerías.....	65
5.2.2	Hojas de estilo	65
5.3	Código de desarrollo	65
5.3.1	Conexión a la base de datos.....	65
5.3.2	Manejo de funciones por versión PHP	66
5.3.3	Inicios de sesión	68
5.3.4	Manejo de Menús a presentar por tipo de usuario.....	70
5.3.5	Manejo de ficheros	71
5.3.6	Consultas básicas a la base de datos.....	73
5.3.7	Restricción de páginas	73
5.4	PROTOTIPADO Y DISEÑO	75
6	CAPÍTULO V: Pruebas de funcionamiento.....	80
7	CAPÍTULO VI: Conclusiones, recomendaciones y anexos.....	80
7.1	Conclusiones.....	80
7.2	Recomendaciones	80
7.3	Anexos	81
7.3.1	Anexo 1: Diagnóstico de procesos de la Dirección General Administrativa 81	
7.3.2	Anexo 2: Script de generación de la base de datos.....	86
7.3.3	Anexo 3: Encuesta	115
7.3.4	Anexo 4: Manual técnico.....	117
7.3.5	Anexo 5: Manual Usuario	117
8	Bibliografía.....	118

2 CAPÍTULO I: INTRODUCCIÓN

2.1 SITUACIÓN ACTUAL

2.1.1 La Universidad

En la actualidad las personas que deseen hacer uso de los parqueaderos en la Pontificia Universidad Católica del Ecuador, deben ser asignados mediante un procesos de selección una plaza de parqueo, este cupo no asegura que siempre exista plazas disponibles, más asegura la entrada libre al campus por medio de un adhesivo entregado, Este adhesivo es revisado por los guardias de seguridad que controlan el uso de los parqueaderos de la PUCE.

La universidad cuenta con 856 plazas, las cuales 774 se encuentran situadas en el campus y 95 en el Girón de estos 622 están disponibles, se tiene 152 plazas ya asignadas, 4 para las personas minusválidas, 9 para las personas que son parte de la residencia Jesuita ubicados al frente del estadio de la Universidad, el resto está dividido entre administrativos y docentes.

Al proceso de selección no ingresan los administrativos ni docentes para lo cual los parqueaderos serán distribuidos entre estudiantes, para el presente semestre se registró 709 estudiantes de pregrado y 804 de postgrado que entraron el proceso de selección.

A la final, se asignaron para estudiantes, 200 parqueaderos en el tipo abierto que no tiene límite de tiempo ni horario de restricción, en el tipo nocturno se asignaron 100 plazas, para el uso único de parqueadero en la zona llamada Girón se asignaron 100 plazas y a estudiantes de postgrado se asignó un total de 300 plazas como único uso los días viernes y sábados.

Las plazas de parqueo no están numeradas y el uso de estas no tiene un control de por medio, ya que no podríamos saber el número de plazas disponibles y no disponibles, esto conlleva a una acumulación innecesaria de vehículos ya que no se sabe cuáles son las plazas y peor los sectores donde se encuentran estas plazas para que puedan ser utilizadas por los estudiantes.

El mal uso de las plazas es otro problema ya que los estudiantes no parquean bien sus vehículos impidiendo el ingreso de otros vehículos a las plazas que se encuentran alrededor.

Los usuarios han encontrado un malestar común. Durante las horas con más recurrencia en la universidad, resulta muy complicado encontrar un espacio disponible para parquear. Por ejemplo, si un estudiante tiene clases a las 9 de la mañana (una de las horas con más aglomeración de vehículos), no podrá encontrar parqueadero si no llega antes de las 7 y media, ya que los parqueaderos se encuentran ocupados. En caso de llegar después de que no haya plazas disponibles, el estudiante podría esperar por un parqueadero entre cinco minutos y una hora que es el promedio por hora de clase. En este

caso el estudiante o profesor se atrasa a clases o en el peor de los casos, falta a las mismas por no tener un lugar donde estacionar.

En muchos casos los estudiantes, profesores o administrativos que ocupan los parqueaderos, no pueden llegar antes al lugar porque viven lejos o por el tráfico pesado de la ciudad a dichas horas. Muchas de estas personas utilizan su vehículo por la misma razón, porque por la distancia, se les complica tomar otro medio de transporte.

Otro agravante, es que muchas personas suelen dejar sus vehículos en la universidad aun cuando no están ahí, para evitar gastos de transporte externos y que también no desean perder el espacio que tienen por lo que no sacan sus vehículos.

2.1.2 Análisis del proceso “ENTREGA DE ADHESIVOS DE PARQUEADERO”.

Análisis (Anexo1):

El proceso inicia a partir de una solicitud que hacen los estudiantes para obtener una plaza de parqueo en los diferentes sectores , los estudiantes ,Personal Docente y Administrativo que resulten favorecidos con el parqueadero tendrán la obligación de colocar los adhesivos que identifiquen al vehículo y a la persona que lo conduce.

Cada una de las fases del proceso están detalladas en el [anexo1](#).

2.2 IDENTIFICACIÓN DEL PROBLEMA

Este proyecto fue ideado para mejorar y optimizar el manejo de parqueaderos en la Pontificia Universidad Católica del Ecuador por medio de un aplicativo web. Esto permitirá a los usuarios reservar un espacio a una hora determinada en la institución para

evitar la espera, mejorar el control que se lleva de los vehículos que ingresan a la institución.

2.3 ALCANCE Y LIMITACIONES

2.3.1 Alcance

El proyecto de implementación tiene como alcance automatizar el proceso de uso de parqueaderos en la Pontificia Universidad Católica del Ecuador en todo el campus.

Cada uno de los aspectos tratados comprenden datos en cuanto a investigaciones realizadas en sobre el uso del parqueaderos en la universidad, se abarcaran temas como el nivel de satisfacción, señalización, seguridad, control de ingreso y de salida.

No se mencionará a profundidad temas como, costo beneficio de la implementación ni proyectos de adecuación y o expansión del sistema.

2.3.2 Limitaciones

Las presentes limitaciones restringirán el avance del proyecto:

La veracidad y fidelidad de los datos que arrojen las encuestas.

Los tiempos de respuesta dependerán del tráfico de solicitudes enviadas al servidor web.

El sistema no estará ligado ni alojado en herramientas que pertenezcan a la universidad por tratarse de un sistema piloto y por salvaguardar la seguridad de la información de la entidad.

2.4 ANÁLISIS DE MERCADO

Entendiendo las condiciones del mercado:

Primero es importante definir cuál es el mercado en el que nos vamos enfocar.

En nuestro caso, el mercado se delimitará a los usuarios que hayan conseguido un cupo para los parqueaderos de la universidad y que hayan cancelado el valor del mismo en la matrícula. Estos usuarios serían los alumnos de pregrado y posgrado que hayan sido favorecidos después del sorteo de cupos. Además consideramos para el estudio a los profesores y administrativos que también utilizan las instalaciones de la universidad para parquear sus vehículos. Se excluirá a todos los espacios reservados o fijos de autoridades que tienen su espacio propio. Este estudio cubre únicamente la sede de Quito de la universidad y el sistema estará planeado para funcionar los 365 días del año.

Considerando que es una institución educativa de nivel superior, podemos asegurar que los usuarios son mayores de edad y se encuentran en promedio entre los 18 y los 65 años, equitativamente distribuidos entre hombres y mujeres. Se estima que la mayoría de los estudiantes que asisten a la universidad poseen un teléfono inteligente y tienen acceso a internet (al menos dentro del campus con la red Wi-Fi de la institución).

Identificar Oportunidades del mercado:

- La mayoría de los usuarios tienen acceso a internet desde sus teléfonos que nos permitiría ofrecer el servicio vía web.
- La universidad ofrece WiFi dentro del campus donde todos los estudiantes, administrativos y profesores tienen acceso desde sus computadoras, móviles, tablets, etc. Esto permite acceder al sistema únicamente con conexión a internet.

- Con el crecimiento de usuarios con smartphones, se cubre cada vez más usuarios que pueden entrar a la aplicación en tiempo real.
- Los usuarios que ya cancelaron el valor del parqueadero tendrán acceso a la aplicación y como el control de stickers se valida en la garita, no habría problemas en el momento de querer hacer reservas externas para las plazas.

2.4.1 Encuesta

La encuesta se encuentra adjunta en el Anexo 3. Se realizó con la ayuda de la herramienta digital gratuita llamada Survey Monkey.

La encuesta se realizó compartiendo a los diferentes usuarios de la universidad (administrativos, estudiantes, profesores) el siguiente link:

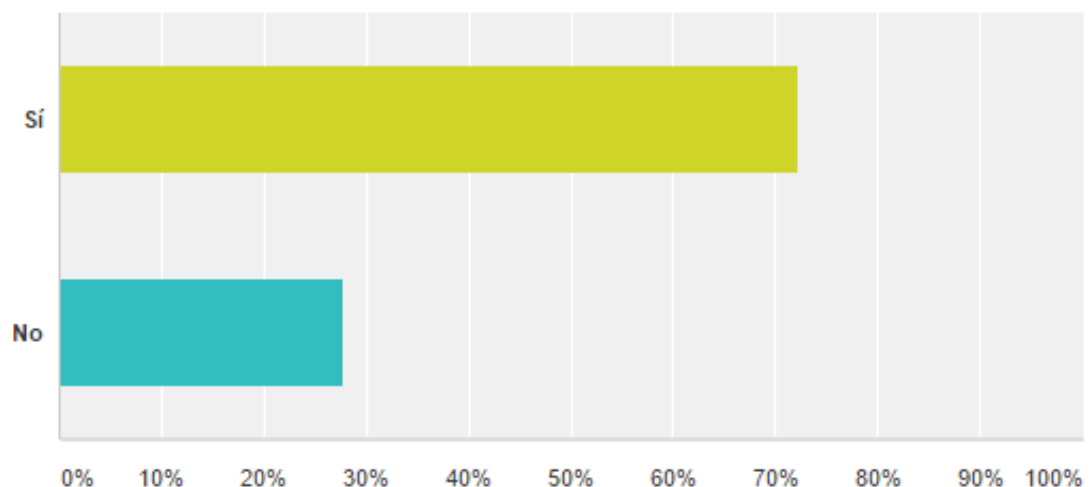
<https://es.surveymonkey.com/r/H55PR8P>

2.4.2 Tabulación y análisis

A continuación se muestra la tabulación y análisis por cada una de las preguntas:

¿Usted posee o ha poseído un parqueadero de la PUCE?

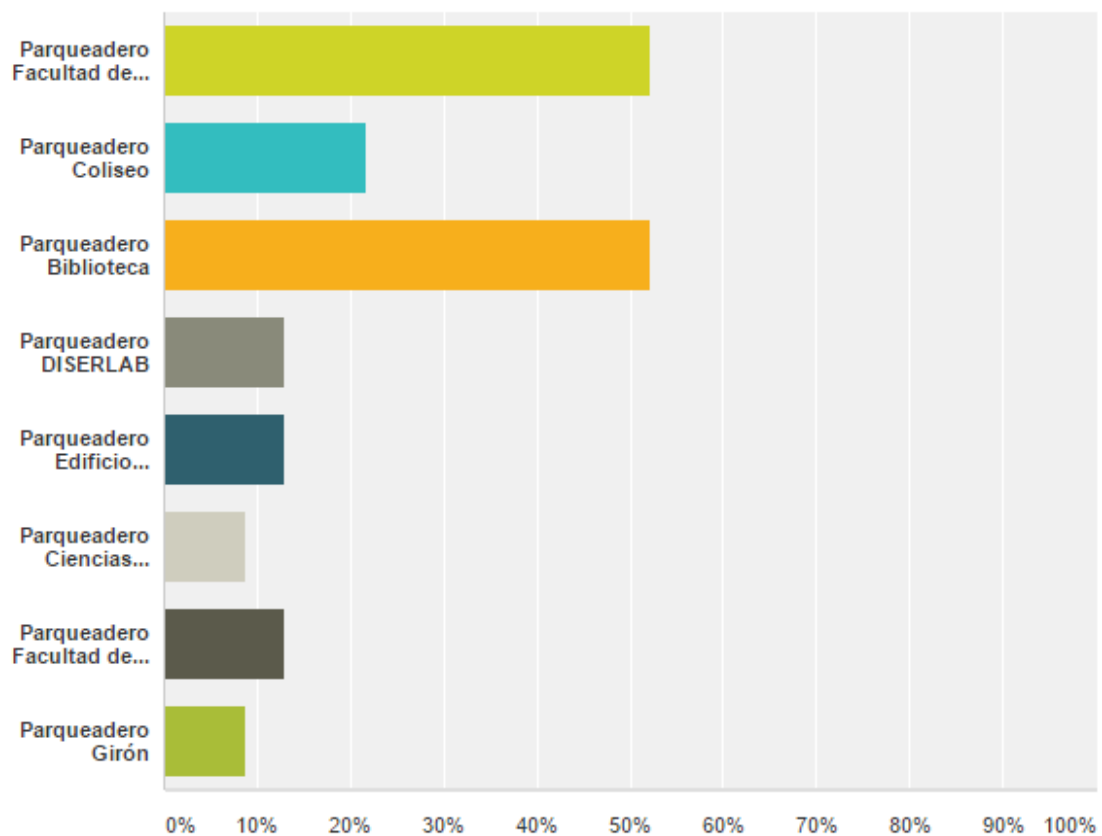
Respondido: 36 Omitido: 0



De las personas que llenaron la encuesta, en su totalidad usuarios de la PUCE, el 72.22% son personas que tienen o han tenido en algún momento parqueadero. Este otro 27.68% no es descartable ya que en el futuro pueden utilizar los parqueaderos en la universidad. Este porcentaje realmente es significativo ya que los usuarios conocen del uso de los parqueaderos y serían nuestros usuarios finales para el aplicativo.

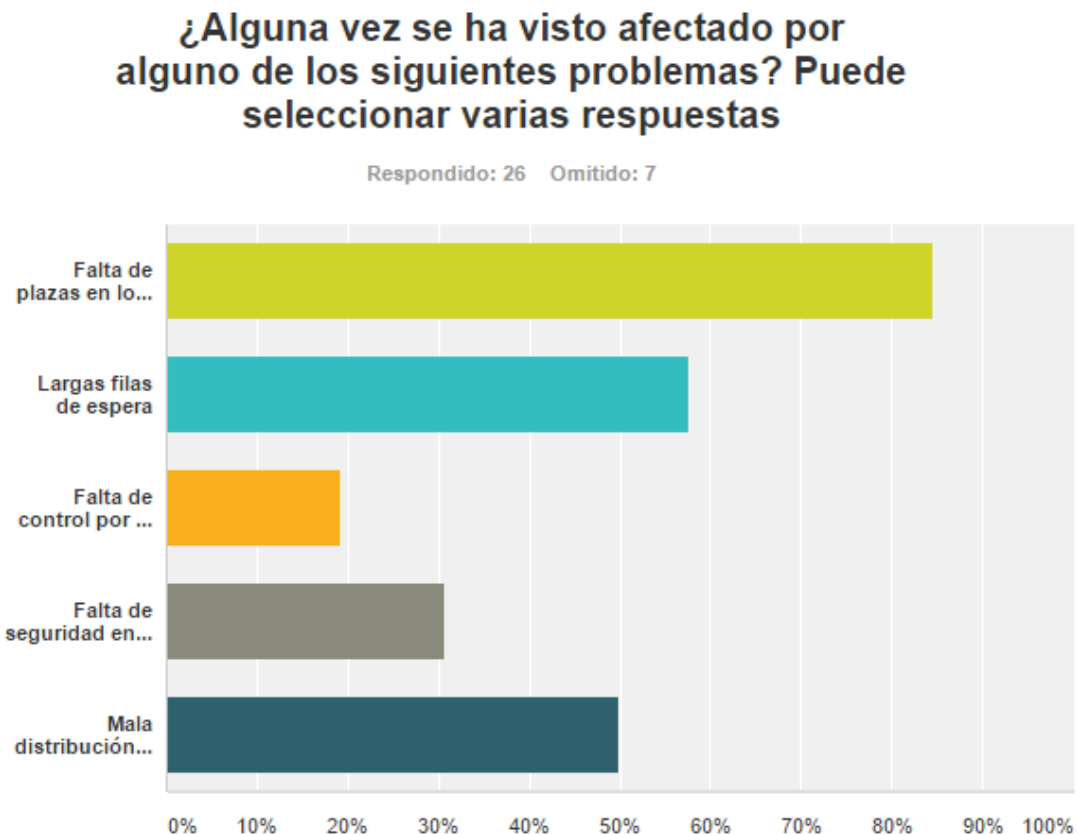
¿Qué sectores ha tenido asignados para su vehículo?

Respondido: 23 Omitido: 10



En su mayoría, las personas encuestadas respondieron que solían tener asignados plazas de parqueaderos en la facultad de Ingeniería y en el sector de la Biblioteca de la universidad. Los parqueaderos con menos asignaciones fueron los de los sectores de Ciencias Exactas y la zona del parqueadero Girón. Esto implicaría que los parqueaderos

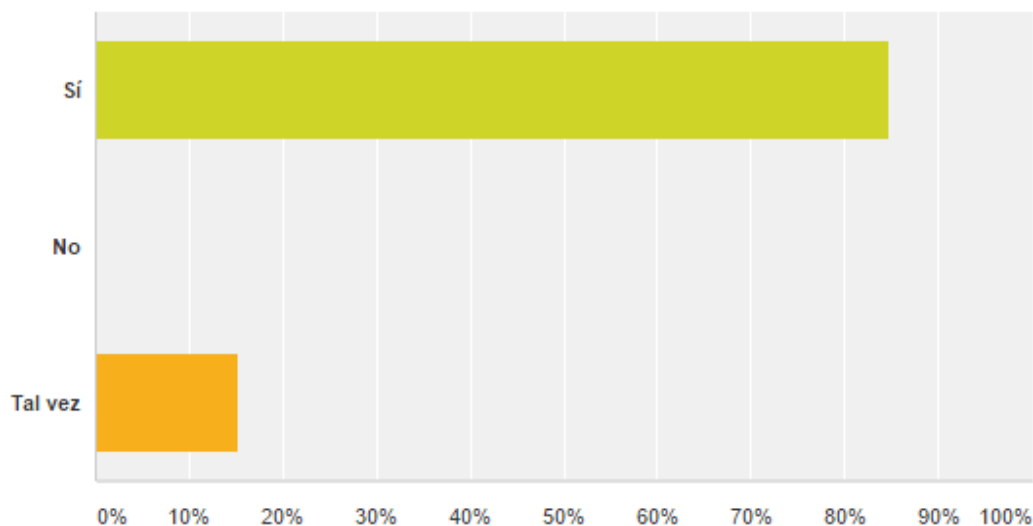
de mayor afluencia, tendrían la mayor cantidad de reservas con el sistema (Fac. Ingeniería y Biblioteca).



El problema más grande identificado en los parqueaderos por los usuarios es la falta de plazas de parqueo en las zonas de la universidad lo que causa molestia en los usuarios y también encontramos que la gente se molesta por las largas filas de espera al no encontrar un parqueadero cuando llegan a la universidad. Con nuestro sistema de reservas se resuelven los dos problemas más grandes que encuentran los usuarios ya que desde el aplicativo, los usuarios pueden ver las plazas disponibles y reservar los espacios vacíos que deseen, además de evitar las largas filas de espera.

¿Estaría dispuesto a utilizar un aplicativo web para reservar un espacio en el parqueadero?

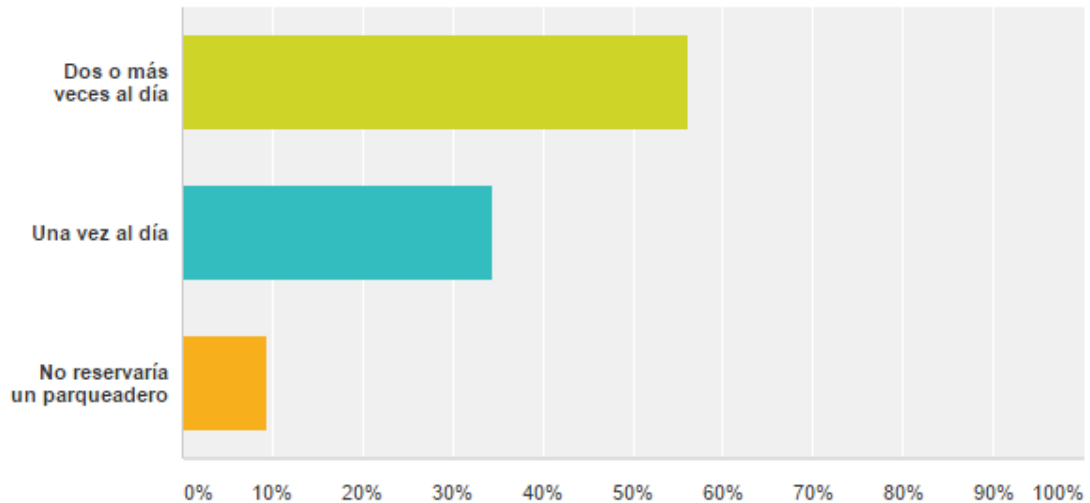
Respondido: 33 Omitido: 0



El 86.11% de las personas encuestadas dijeron que si utilizarían a aplicación de reservas y apenas el 13.89% dijo que tal vez. En contra de lo esperado, ninguna de las personas encuestadas respondió que no utilizaría el aplicativo, lo cual es una buena noticia ya que todos los usuarios podrían o estarían dispuestos a intentar usar nuestro sistema. Podría ser que el porcentaje que respondió que tal vez utilizaría el sistema, tiene dudas del funcionamiento de la aplicación o tal vez no tiene acceso a un teléfono con conexión a internet para poder realizar las reservas del parqueadero.

¿Con que frecuencia reservaría parqueadero en la universidad por día?

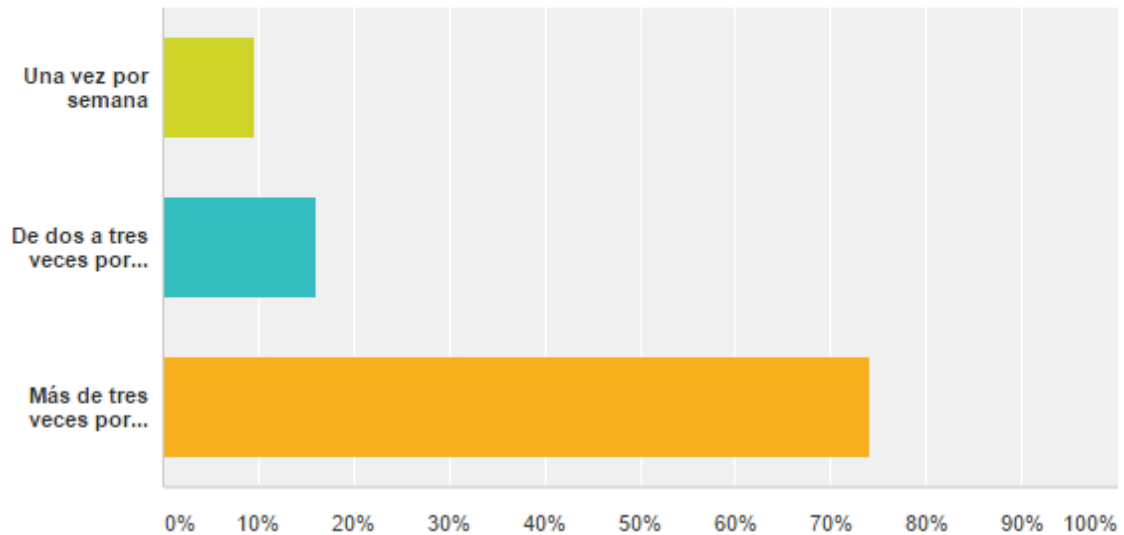
Respondido: 32 Omitido: 1



El 54.39%, respondió que realizaría más de dos reservar al día del parqueadero, lo que indica que más de la mitad de usuarios utilizarían nuestro servicio cada vez que requieran ingresar con su vehículo a la universidad. El 37.14% utilizaría las reservas del sistema al menos una vez al día lo que indica que más del 91% definitivamente realizaría la reserva de una plaza para parquear a diferencia del 9 % que no realizaría la reserva.

¿Con que frecuencia por semana reservaría parqueadero en la universidad?

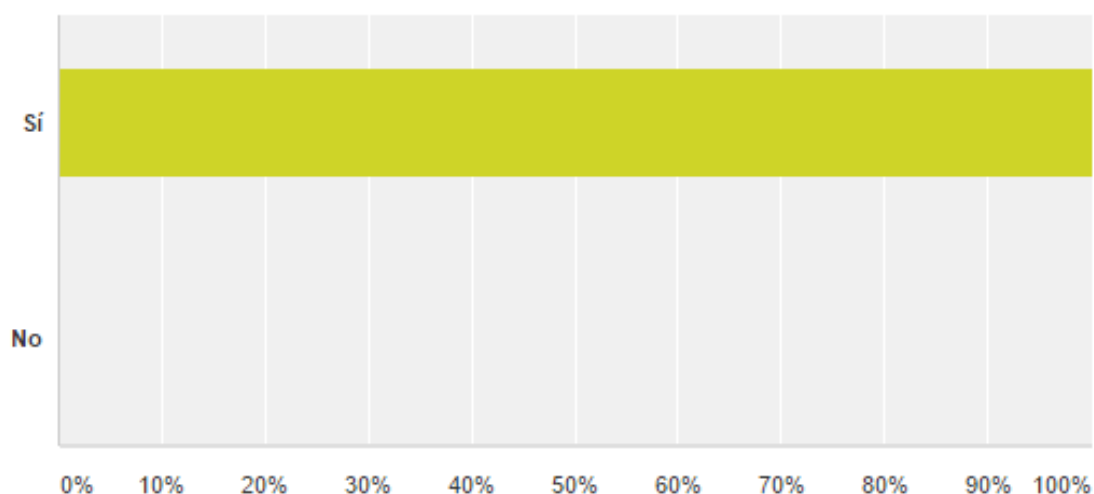
Respondido: 31 Omitido: 2



Considerando que la mayoría de profesores, administrativos y estudiantes asisten la mayor parte de la semana, no es sorpresa que más del 70% de los encuestados realicen reservas más de tres veces por semana y la diferencia en los que no usarían el servicio de las reservas, no representa un porcentaje muy significativo en la universidad, pero deben ser considerados de todas formas.

¿Desearía encontrar datos sobre las infracciones cometidas en el aplicativo de reservas?

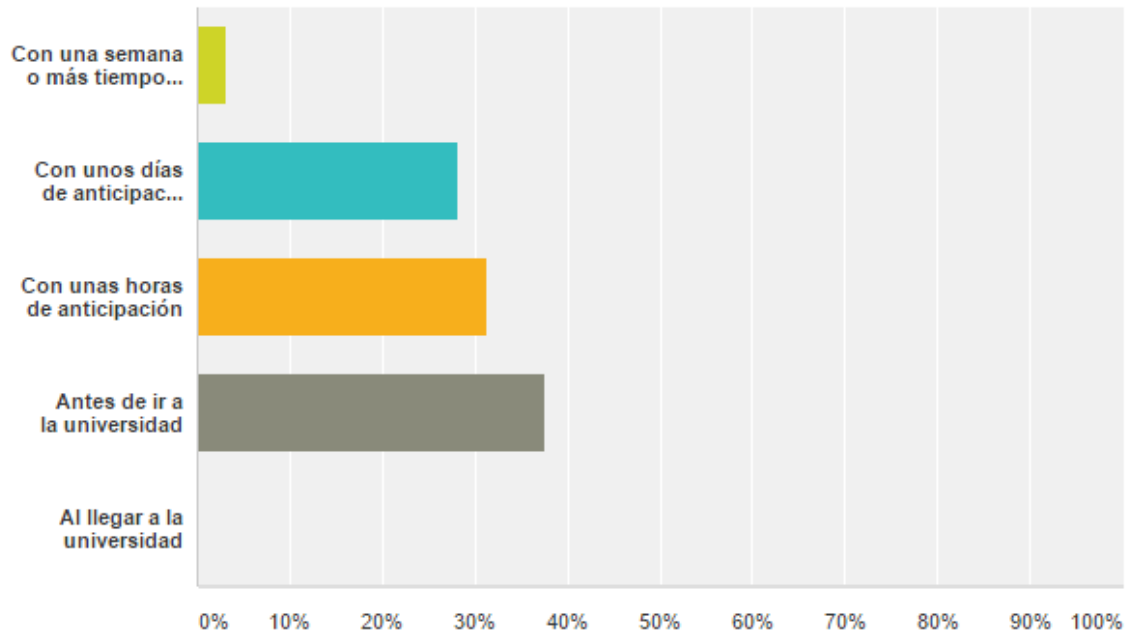
Respondido: 33 Omitido: 0



Como se observa en la gráfica, todos los usuarios estarían de acuerdo con poder visualizar las infracciones cometidas en la aplicación ya que facilita la consulta de las mismas y pueden consultarse en cualquier momento y en cualquier lugar.

¿Con cuánto tiempo de anticipación reservaría una plaza en los parqueaderos?

Respondido: 32 Omitido: 1



Como vemos, la mayoría de personas realizaría las reservas antes de entrar a la universidad, lo que significa que es más práctico que la aplicación sea web y funcione a tiempo real, como estamos presentando en nuestro sistema.

2.4.3 Técnicas de análisis

Técnicas de análisis de datos:

- Técnica para la recolección de datos: Encuesta por medio de Survey Monkey enfocada a los usuarios del parqueadero de ingeniería de la PUCE.
- Técnica de costo beneficio: Analizaremos los beneficios de la implementación del sistema en comparación con los costos para generar retorno de la inversión

3 CAPÍTULO II: MARCO TEÓRICO

3.1 HERRAMIENTAS DE MODELADO

3.1.1 Metodología de desarrollo UWE.

3.1.1.1 Definición

UWE es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las técnicas de UML, la notación de UML y los mecanismos de extensión de UML.

Es una herramienta que nos permitirá modelar aplicaciones web, utilizada en la ingeniería web, prestando especial atención en sistematización y personalización (sistemas adaptativos). UWE es una propuesta basada en el proceso unificado y UML pero adaptados a la web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito.

En el marco de UWE es necesario la definición de un perfil UML (extensión) basado en estereotipos con este perfil se logra la asociación de una semántica distinta a los diagramas del UML puro, con el propósito de acoplar el UML a un dominio específico, en este caso, las aplicaciones Web. Entre los principales modelos de UWE podemos citar: el modelo lógico-conceptual, modelo navegacional, modelo de presentación, visualización de Escenarios Web y la interacción temporal, entre los diagramas: diagramas de estado, secuencia, colaboración y actividad.

UWE define vistas especiales representadas gráficamente por diagramas en UML. Además UWE no limita el número de vistas posibles de una aplicación, UML proporciona mecanismos de extensión basados en estereotipos. Estos mecanismos de extensión son los que UWE utiliza para definir estereotipos que son lo que finalmente se utilizarán en las vistas especiales para el modelado de aplicaciones Web. De esta manera, se obtiene una notación UML adecuada a un dominio en específico a la cual se le conoce como Perfil UML.

UWE está especializada en la especificación de aplicaciones adaptativas, y por tanto hace especial hincapié en características de personalización, como es la definición de un modelo de usuario o una etapa de definición de características adaptativas de la navegación en función de las preferencias, conocimiento o tareas de usuario.

Además de estar considerado como una extensión del estándar UML, también se basa en otros estándares como por ejemplo: XMI como modelo de intercambio de formato, MOF para la meta-modelado, los principios de modelado de MDA, el modelo de transformación del lenguaje QVT y XML.

3.1.1.2 Actividades de modelado de UWE

Las actividades base de modelado de UWE son el análisis de requerimientos, el modelo conceptual, el modelo de navegación y el modelo de presentación. A estos modelos se pueden sumar otros modelos como lo son el modelo de interacción y la visualización de Escenarios Web.

3.1.1.3 Etapas de modelado

El modelo que propone UWE está compuesto por etapas o sub-modelos:

- Modelo de Requerimientos
 - Definir Actores

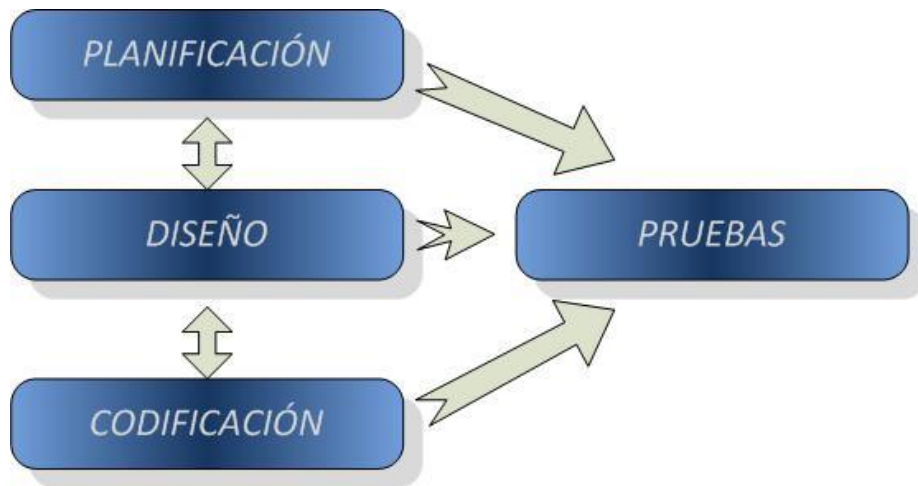
- Casos de uso de la aplicación y sus relaciones
 - Actividades describiendo los casos de uso en detalle
- Modelo de Contenido
- Modelo de Navegación
- Modelo de Presentación
- Modelo de Procesos

3.1.1.4 FASES de la metodología UWE

UWE cubre todo el ciclo de vida de este tipo de aplicaciones centrando además su atención en aplicaciones personalizadas o adaptativas.

Las fases o etapas a utilizar son:

- 1) Captura, análisis y especificación de requisitos.
- 2) Diseño del sistema:
- 3) Codificación del software:
- 4) Pruebas: Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código.
- 5) La Instalación o Fase de Implementación
- 6) El Mantenimiento



3.2 HERRAMIENTAS DE DESARROLLO DE SOFTWARE

3.2.1 Herramientas de diseño

3.2.1.1 *MagicDraw*

3.2.1.1.1 Introducción

Magic Draw es una herramienta para modelado de procesos, arquitectura de software y sirve como sistema de apoyo en el trabajo en equipo. Fue diseñado por programadores, ingenieros de control de calidad, analistas de negocios y analistas de software para facilitar el análisis y diseño de los sistemas orientados a objetos y las bases de datos. Proporciona mejor mecanismo de la industria de la ingeniería de código (con el apoyo de ida y vuelta completo para Java, C ++, C #, CL (MSIL) y lenguajes de programación CORBA IDL), así como el modelado de esquemas de bases de datos, generación de DDL y de instalaciones de ingeniería inversa. (Nomagic.com, 2016)

3.2.1.1.2 Características

- Interfaz elegante e intuitiva, la mayor parte de las opciones accesibles con un solo clic.

- Ayudas en el diseño con autocompletación y corrección automática en tiempo real.
 - Permite visualizar el proyecto de diferentes formas.
 - Posible derivación de modelos UML a través de códigos fuentes escritos anteriormente.
 - Facilidad y rapidez para el cambio del dominio del modelado.
 - Generador automático de informes.
 - Desarrollo colaborativo directamente con la herramienta a través del Team Work Server(Software que permite trabajar a más de un desarrollador sobre el mismo proyecto en un mismo instante, el modelo está almacenado en un equipo servidor y los desarrolladores pueden consultar y actualizar la información).
 - Disponible para un gran número de plataformas y sistemas operativos.
- (MagicDraw - EcuRed, 2016)

3.2.1.2 *ArgoUML*

3.2.1.2.1 Introducción

ArgoUML es la herramienta de modelado UML código abierto e incluye soporte para todos los diagramas UML 1.4 estándar. Se ejecuta en cualquier plataforma Java y está disponible en diez idiomas. (argouml.tigris.org, 2016)

3.2.1.2.2 Características

- ArgoUML es muy potente y fácil de usar con un entorno interactivo, es sistema soporta diseño, desarrollo y documentación de las aplicaciones de software orientado a objetos.

- Los usuarios de ArgoUML son los diseñadores de software y arquitectos, desarrolladores de software, analistas de negocios, analistas de sistemas y otros profesionales involucrados en el análisis, diseño y desarrollo de aplicaciones de software.
- Los estándares abiertos: XMI, SVG y PGML
- Es 100% independiente de la plataforma gracias al uso exclusivo de Java
- Open Source, que permite la ampliación o personalización del mismo.
- Funciones cognitivas como: la reflexión en la acción, diseño oportunista, la comprensión y resolución de problemas. (Argouml-stats.tigris.org , 2016)

3.2.1.3 PowerDesigner

3.2.1.3.1 Introducción

Es la solución de procesos de negocios líder en la industria, es un software de modelado de datos y gestión de metadatos para la arquitectura de datos, arquitectura de la información y la arquitectura de la empresa. (SAP Sybase PowerDesigner | product information website, s.f.)

3.2.1.3.2 Características

PowerDesigner trae técnicas de análisis de impacto de gran alcance, la gestión del cambio en tiempo de diseño y gestión de metadatos para su empresa.

PowerDesigner está equipado de forma única para apoyar a todos los entornos arquitectónicos utilizando técnicas de modelización y gestión de metadatos combinados. (SAP Sybase PowerDesigner | product information website, s.f.)

El repositorio de metadatos de **PowerDesigner** mejora la colaboración y la comunicación entre todas las partes interesadas en la empresa, lo que facilita una respuesta más rápida a cambiar para mejorar la agilidad del negocio. (SAP Sybase PowerDesigner | product information website, s.f.)

3.2.1.3.3 PowerDesigner ofrece

- El aumento de su productividad
- El apoyo abierto para entornos heterogéneos
- Un entorno altamente personalizable que les apoya en la aplicación de normas y reglamentos
- Más fácil de arquitectura empresarial implementaciones por comprensión intuitiva de todas las intersecciones en todas las áreas de su negocio
- El aumento de su rendimiento empresarial a través de la utilización de Enlace y sincronización para los análisis de impacto verdaderos

3.2.1.4 *MYSQL Workbench*

3.2.1.4.1 Introducción

MySQL Workbench es una herramienta visual unificada para los arquitectos de bases de datos, desarrolladores y administradores de bases. Ofrece modelado de datos, desarrollo de SQL y herramientas de administración integrales para la configuración del servidor, administración de usuarios, copia de seguridad, y mucho más. (MySQL :: MySQL Workbench, 2016)

3.2.1.4.2 Características

- MySQL Workbench está disponible en Windows, Linux y Mac OS X.
- Permite a un DBA, desarrollador o arquitecto de datos diseñar visualmente, modelar, generar y gestionar bases de datos.
- Proporciona herramientas visuales para crear, ejecutar, y optimizar consultas SQL. También proporciona una consola visual para administrar fácilmente entornos MySQL y obtener una mejor visibilidad en bases de datos.
- MySQL Workbench proporciona un conjunto de herramientas para mejorar el rendimiento de las aplicaciones MySQL. Y genera una solución completa y fácil de usar para la migración de Microsoft SQL Server, Microsoft Access, Sybase ASE, PostgreSQL, y otras tablas RDBMS, objetos y datos a MySQL. (MySQL :: MySQL Workbench, 2016)

3.2.2 Herramientas de desarrollo

3.2.2.1 *MYSQL*

3.2.2.1.1 Introducción

MySQL es un sistema de gestión de bases de datos relacional que está considerada como la base datos open source más popular del mundo para entornos de desarrollo web.

MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones.

Los elementos considerados como esenciales fueron incorporados tanto por desarrollos internos, como por desarrolladores de software libre. (MySQL | The Most Popular Open-Source Database | Oracle, 2016)

3.2.2.1.2 Características

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Posibilidad de selección de mecanismos de almacenamiento que ofrecen diferentes velocidades de operación, soporte físico, capacidad, distribución geográfica, transacciones.
- Transacciones y claves foráneas.
- Conectividad segura.
- Replicación.
- Búsqueda e indexación de campos de texto. (Es.wikipedia.org, 2016)

3.2.2.2 *PHP*

3.2.2.2.1 Introducción

PHP es un lenguaje de programación de propósito general popular que es especialmente adecuado para el desarrollo web. (PHP: Hypertext Preprocessor, 2016)

PHP es un lenguaje de script del servidor, y una herramienta poderosa para hacer páginas web dinámicas e interactivas. También es una alternativa eficiente gratuita que se utiliza ampliamente a sus competidores, como ASP de Microsoft. (PHP 5 Tutorial, 2016)

3.2.2.2.2 Características

- Autenticación HTTP con PHP
- 'Cookies'

- Sesiones
- Manejo de XForms
- Subida de ficheros
- Empleo de ficheros remotos
- Manejo de conexiones
- Conexiones persistentes a bases de datos
- Modo seguro
- Recolección de basura
- Rastreo Dinámico con DTrace (Php.net, 2016)

3.2.2.3 Javascript

3.2.2.3.1 Introducción

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas.

Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. (Librosweb.es, 2016)

3.2.2.3.2 Características

- Es simple, no hace falta tener conocimientos de programación para poder hacer un programa en JavaScript.
- Maneja objetos dentro de nuestra página Web y sobre ese objeto podemos definir diferentes eventos. Dichos objetos facilitan la programación de páginas interactivas, a la vez que se evita la posibilidad de ejecutar comandos que puedan ser peligrosos para la máquina del usuario, tales como formateo de unidades, modificar archivos etc.
- Es dinámico, responde a eventos en tiempo real. Eventos como presionar un botón, pasar el puntero del mouse sobre un determinado texto o el simple hecho de cargar la página o caducar un tiempo. Con esto podemos cambiar totalmente el aspecto de nuestra página al gusto del usuario, evitándonos tener en el servidor una página para cada gusto, hacer cálculos en base a variables cuyo valor es determinado por el usuario, etc. (Uzuay.edu.ec, 2016)

3.2.2.4 HTML

3.2.2.4.1 Introducción

HTML es un lenguaje de programación que se utiliza para el desarrollo de páginas de Internet. Se trata de la sigla que corresponde a HyperText Markup Language, es decir, Lenguaje de Marcas de Hipertexto, que podría ser traducido como Lenguaje de Formato de Documentos para Hipertexto. (Definición.de, 2016)

3.2.2.4.2 Características

- HTML es distribuido ya que la información repartida en páginas no muy grandes se encuentran enlazadas entre sí.
- HTML es hipertexto y es fácil navegar por él.

- HTML es compatible con todo tipo de ordenadores (PCs, Macintosh, estaciones de trabajo...) y con todo tipo de sistemas operativos (Windows, MS-DOS, UNIX, MAC-OS,...).
- HTML es dinámico por que el proceso de cambiar y actualizar la información debe ser ágil y rápido. (iBiblio, 2016)

3.2.2.5 *AJAX*

3.2.2.5.1 Introducción

AJAX, son las siglas de Asynchronous JavaScript And XML. AJAX es una técnica de desarrollo web que permite actualizar los procesos independientemente de los otros componentes que tiene, intercambiando la información entre el cliente final y un servidor en que se aloja la aplicación. (W3schools.com, 2016)

3.2.2.5.2 Características

- Enviar información a un server que aloja la aplicación desde un cliente que se conecta a un servidor. (Es.wikipedia.org, 2016)
- Actualizar una página sin la necesidad de recargarla y solicitar y recibir datos desde el servidor. (SearchNetworking, 2016)

3.2.2.6 *Bootstrap*

3.2.2.6.1 Introducción

Bootstrap es un framework que funciona con HTML, CSS y JS que permite desarrollo receptivo enfocado en móviles. Es Open-Source, hosteado, desarrollado y mantenido en Github.

3.2.2.6.2 Características

- Utiliza los preprocesadores de Less y Sass.

- Se puede utilizar inmediatamente ya que tiene CSS previamente compilados y contruidos en la fuente.
- Un solo framework multiplataforma que funciona en todo tipo de dispositivos.
- Consta de componentes únicos y totalmente compatibles con HTML.
- Trae varios plugins jQuery multifuncionales. (Otto, 2016)

3.2.2.7 XAMPP

3.2.2.7.1 Introducción

XAMPP es un servidor que no necesita plataforma, que tiene entorno de desarrollo hecho con PHP, de distribución gratuita por parte de Apache. Contiene MariaDB, PHP, Perl. (Apachefriends.org, 2016)

3.2.2.7.2 Características

- Gratis y fácil de instalar.
- Gestión de sistema de bases de datos MySQL.
- Servidor Web Apache. (Es.wikipedia.org, 2016)

3.2.2.8 Apache

3.2.2.8.1 Introducción

Apache es un servidor Web, desarrollado por Apache Software Foundation establecido en 1999. Es fácil de utilizar, muy seguro y fácil de utilizar. Puede ser personalizado dependiendo de las necesidades en diferentes entornos con diferentes módulos y extensiones. (Wpbeginner.com, 2016)

3.2.2.8.2 Características

- Open Source
- Gratis y fácil de utilizar

- Corre más del 67% de los servidores Web en el mundo (Apache.org, 2016)

3.2.2.9 *Ripple*

3.2.2.9.1 Introducción

Ripple es una extensión para Google Chrome que funciona como emulador de un navegador para un dispositivo móvil. Sirve para el desarrollo y pruebas de aplicativos web que son adaptables con diferentes equipos móviles.

3.2.2.9.2 Características

- Multiplataforma
- Enfocado en HTML5
- Ofrece a interfaz de un dispositivo móvil desde el explorador de una PC.
- Permite debugging con Javascript, inspección DOM para HTML.

(Chrome.google.com, 2016)

3.2.2.10 *Sublime Text 3*

3.2.2.10.1 Introducción

Sublime es un editor de texto que se puede descargar gratuitamente, consta con una interfaz amigable y personalizado. Posee varias funcionalidades como multicursores, búsqueda dinámica, soporte de plugins, entre otras.

3.2.2.10.2 Características

- Permite visualizar varios tipos de archivos con diferentes extensiones.
- Tiene atajos que permiten un manejo sencillo uso del aplicativo.
- Permite edición de varias líneas a la vez. (SublimeText, 2016)

3.2.2.11 JQuery

3.2.2.11.1 Introducción

Es una librería de JavaScript, que permite una sencilla manipulación de código para que sea adaptable a varios navegadores, que además facilita la interacción con HTML. Funciona como un fichero único que permite cambiar una página web sin que sea necesario volver a cargarla. (jQuery.com, 2016)

3.2.2.11.2 Características

- Software libre y tiene código abierto.
- Permite selección de elementos DOM.
- Permite manipular las extensiones y hojas de estilo (Es.wikipedia.org, 2016)

3.2.2.12 CSS

3.2.2.12.1 Introducción

Cascade Style Sheet, CSS es un lenguaje que tiene la información relacionada al estilo de una página HTML o XML y explica cómo cada objeto debe ser mostrado. Fueron creados con el objetivo de separar el contenido y funcionamiento de una página con su estilo. (W3schools.com, 2016)

3.2.2.12.2 Características

- Permite una centralización de todos los estilos utilizados en una página, lo que facilita la edición de la misma.
- Optimiza el ancho de banda de la conexión, ya que se pueden hacer hojas de estilos por diferentes secciones.
- Mejor accesibilidad en todo el aplicativo web. (Es.wikipedia.org, 2016)

4 CAPÍTULO III: DISEÑO DE LA APLICACIÓN

4.1 MODELO DE REQUERIMIENTOS

4.1.1 Definir actores

Definición de actores del sistema para reserva de parqueaderos en la Pontificia Universidad Católica del Ecuador.

Se definen varios tipos de actores:

4.1.1.1 *Usuario*

Se define como usuario a aquel actor que va a ingresar al sistema mediante por medio de su usuario y su clave, el usuario será capaz de generar una reserva para un horario específico así como también cancelar la reserva realizada.

4.1.1.2 *Administrador Master*

El administrador Master es el actor capaz de manejar casi todas las funcionalidades del sistema.

4.1.1.3 *Guardia de Control*

Es aquel actor encargado de controlar los accesos y las salidas de cada uno de los vehículos por medio del ingreso al sistema de las placas del vehículo. Así también se encargara de determinar si el vehículo puede o no puede ingresar a los parqueaderos de la universidad.

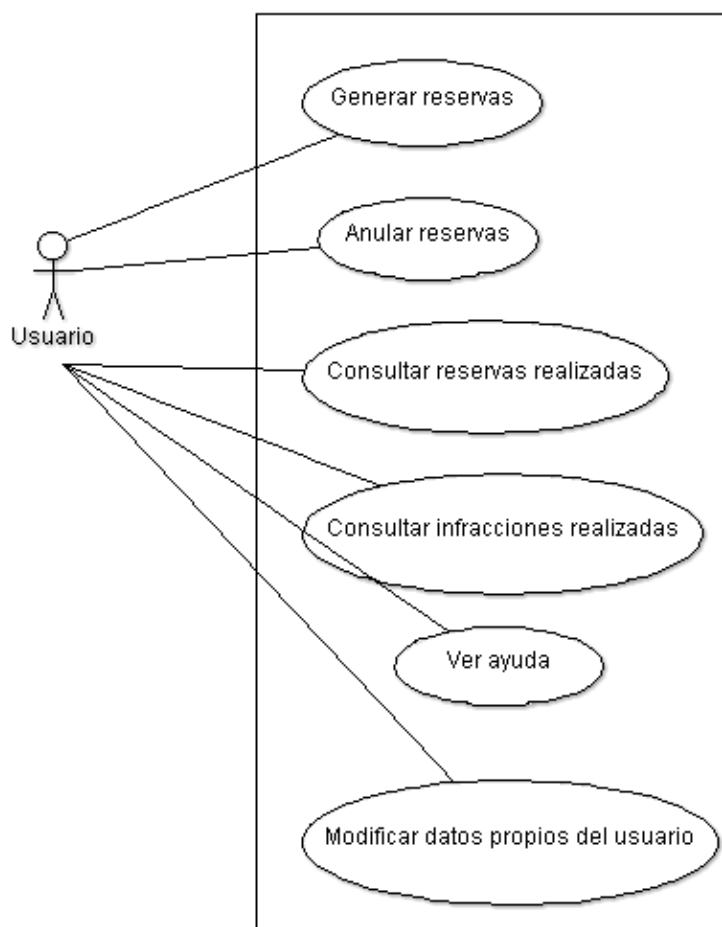
4.1.1.4 *Guardia de Recorrido*

Este actor recorrer los parqueaderos de la universidad determinando, cuál de los vehículos no está cumpliendo con la asignación de reserva del sistema registrando la penalización para determinado usuario.

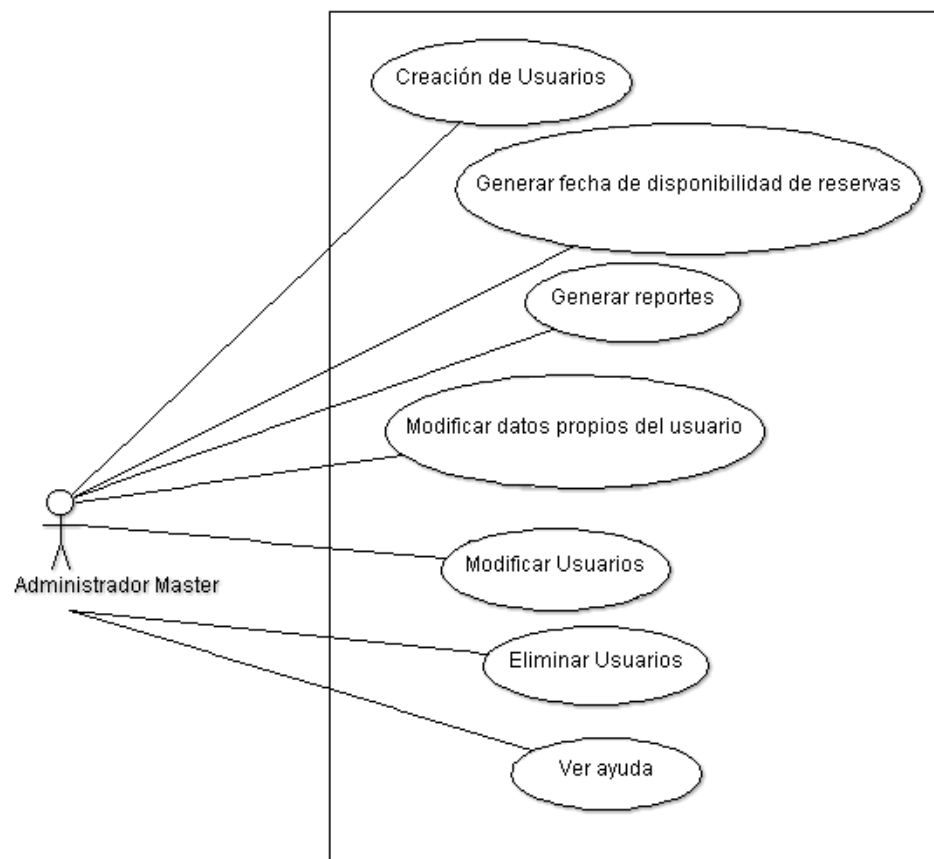
4.1.2 Casos de uso de la aplicación y sus relaciones

Es necesario comenzar modelando los casos de uso de la aplicación para poder modelar las funcionalidades de nuestra aplicación:

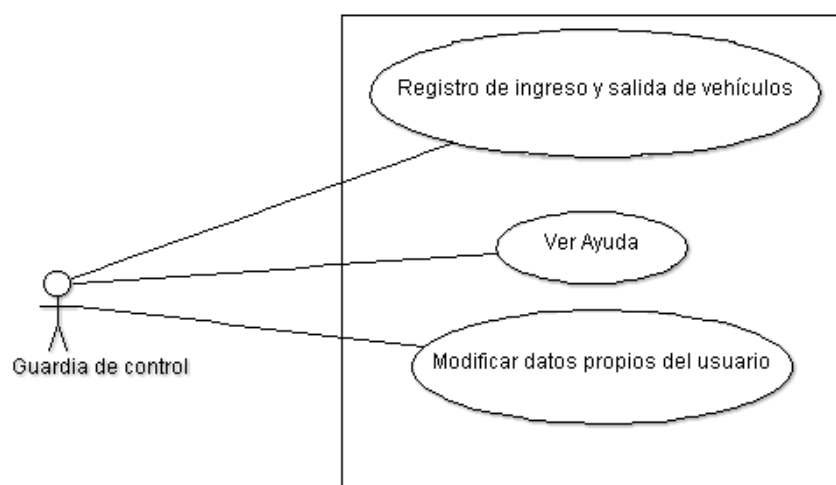
4.1.2.1 Caso de uso para el actor “Usuario”



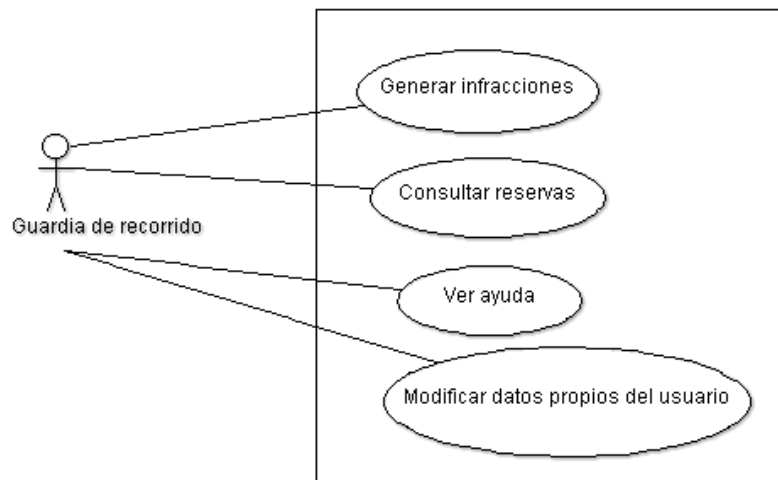
4.1.2.2 Caso de uso para el actor “Administrador Master”



4.1.2.3 Caso de uso para el actor “Guardia de control”

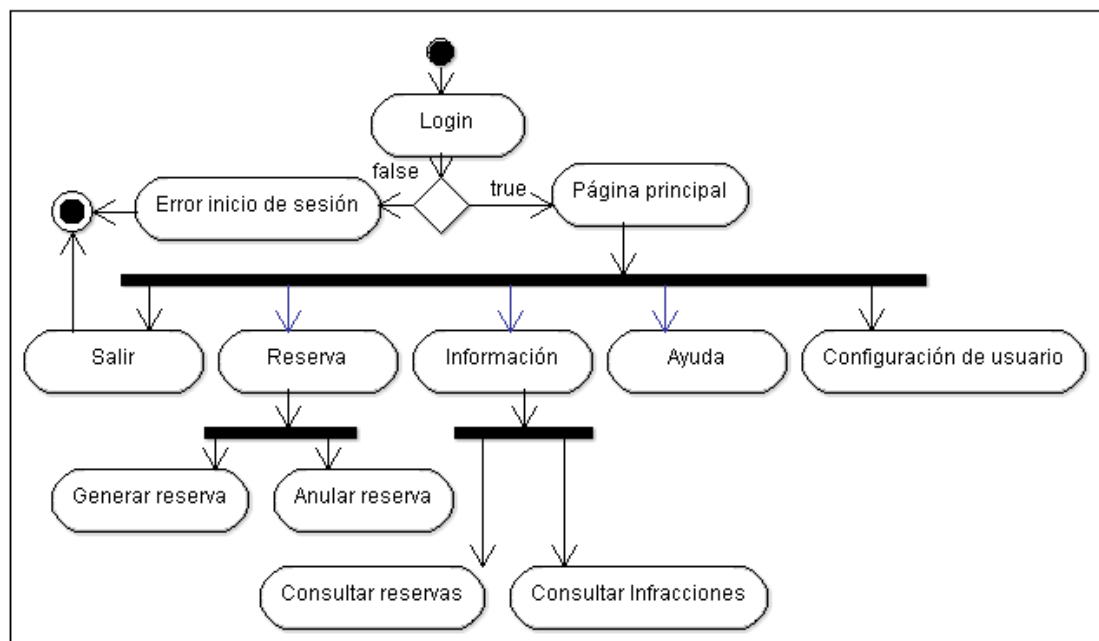


4.1.2.4 Caso de uso para el actor “Guardia de recorrido”

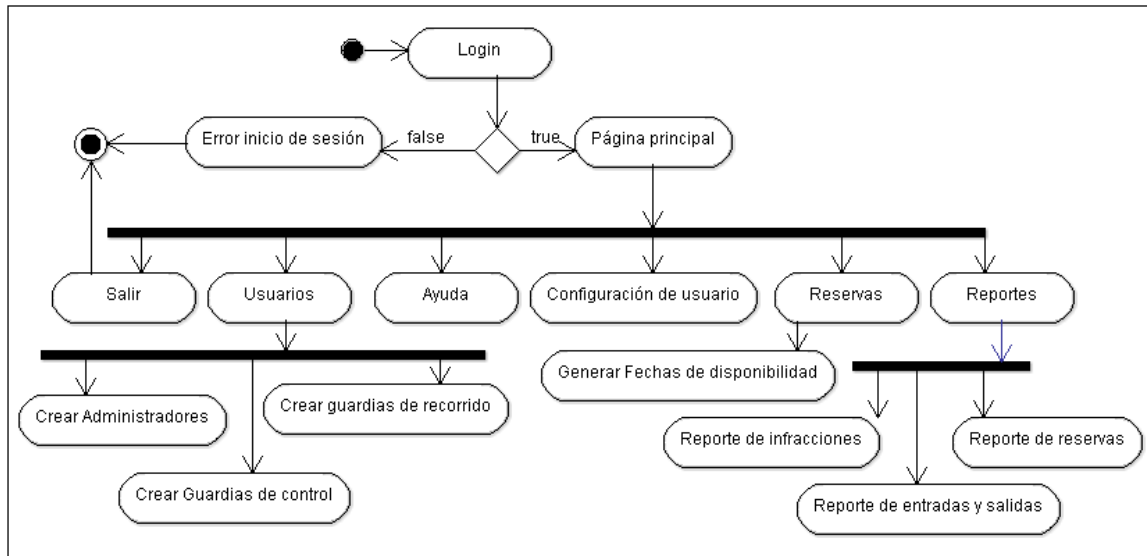


4.1.3 Actividades describiendo los casos de uso en detalle

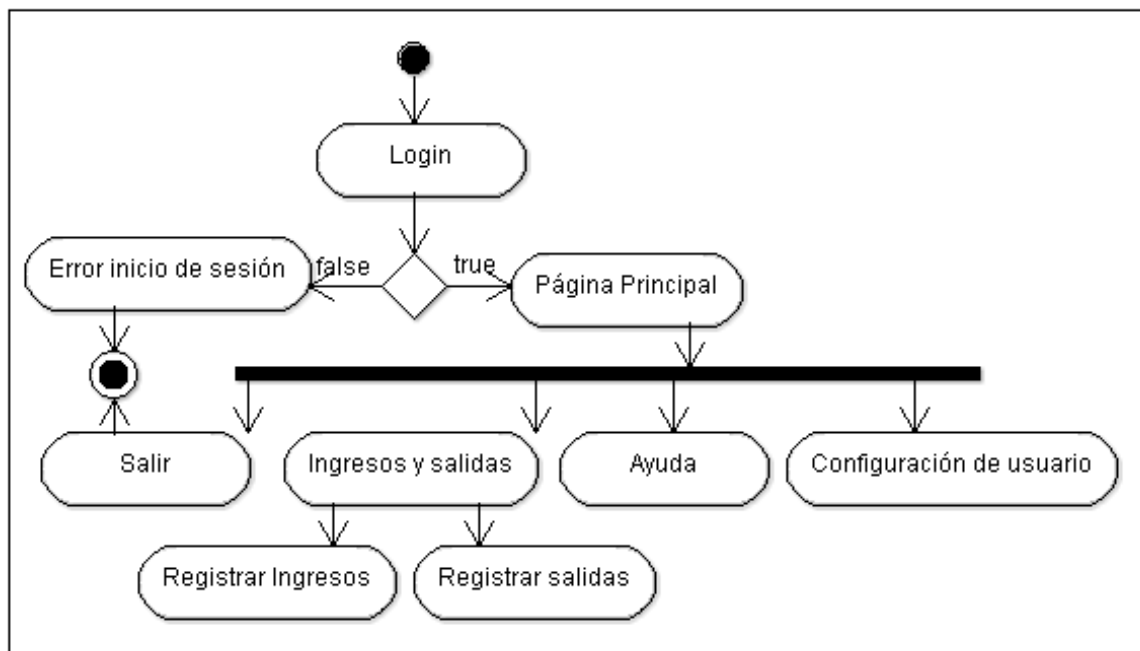
4.1.3.1 Diagrama de actividades para el actor “Usuario”



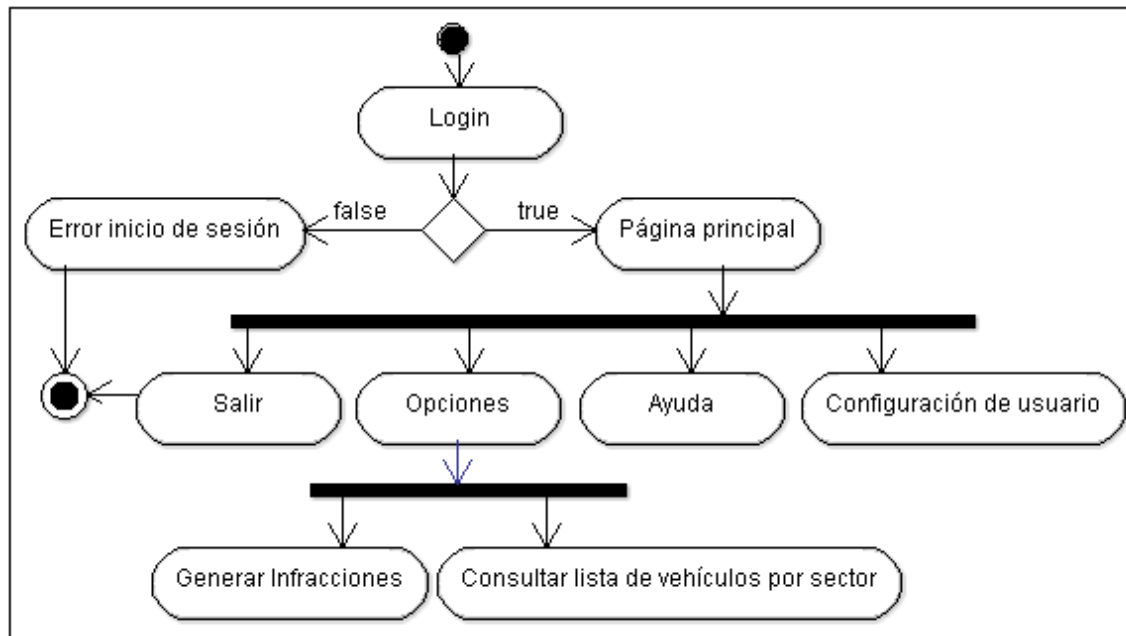
4.1.3.2 Diagrama de actividades para el actor “Administrador master”



4.1.3.3 Diagrama de actividades para el actor “Guardia de control”



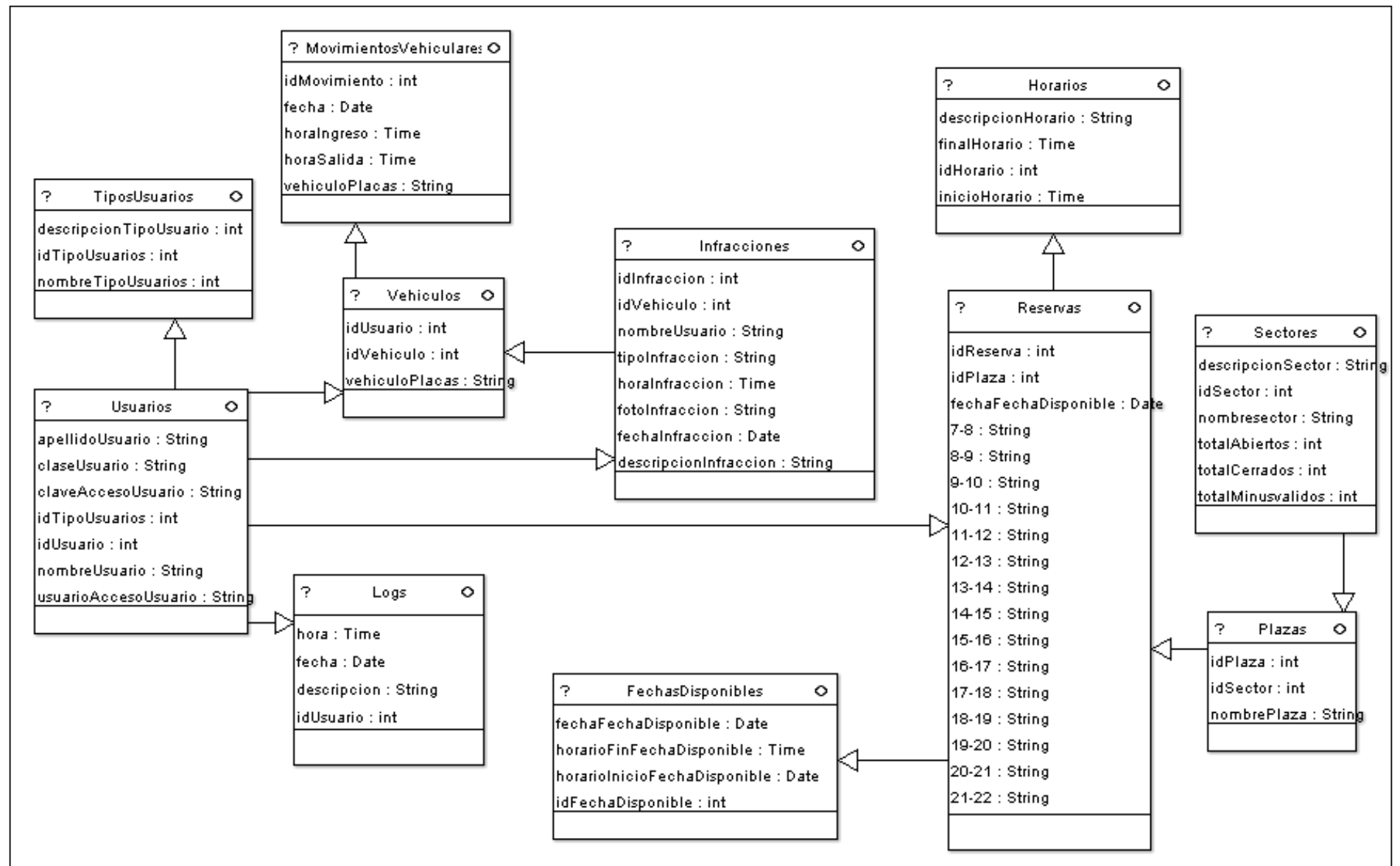
4.1.3.4 Diagrama de actividades para el actor “Guardia de recorrido”



4.2 MODELO DE CONTENIDO

4.2.1 Diagrama de clases

Representaremos cada uno de los grupos que son dados por la base de datos como un diagrama Entidad/Relación.



4.3 MODELO DE NAVEGACIÓN

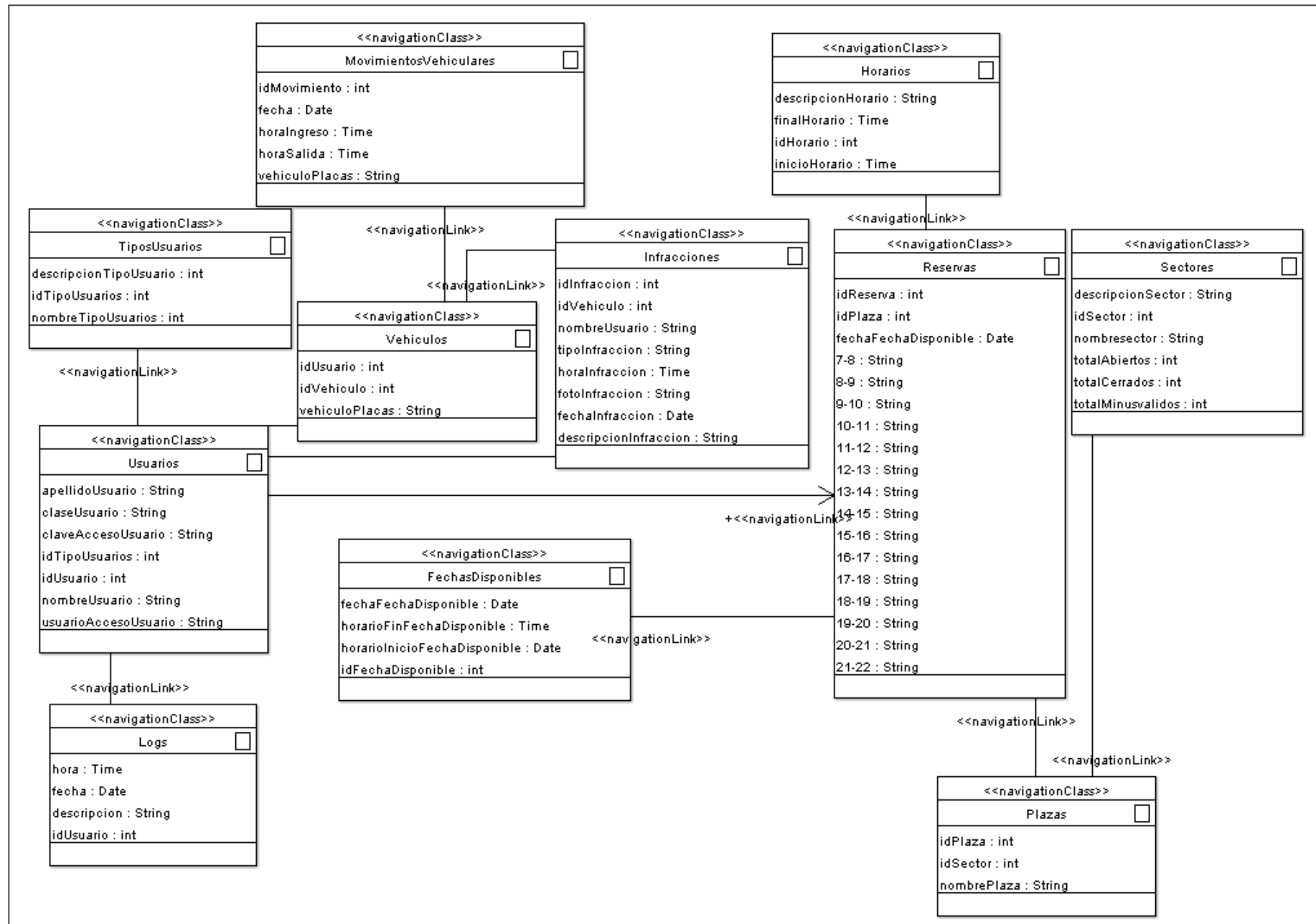
Necesitamos saber cómo se enlazan cada una de las páginas al sistema para ellos crearemos un diagrama que contenga nodos y enlaces.

Nodo es una unidad de navegación que están conectados con enlaces y cada uno de los nodos pueden estar representados por una o varias páginas.

Estereotipos utilizados por UWE:

Nodos: «navigationClass».













Enlaces: «navigationLink».



4.4 MODELO DE PRESENTACIÓN

Ya que los diagramas de navegación no presentan las clases de navegación y los procesos del sistema web, generamos un diagrama de presentación que nos ayuda a generar y expresar cada uno de los elementos y propiedades que pueden indexarse a los objetos del sistema.

nombres de estereotipos y sus iconos

 grupo de presentación	 página de presentación
 texto	 entrada de texto
 ancla	 fileUpload
 botón	 imagen
 formulario	 componente de cliente
 alternativas de presentación	 selección

(Busch, 2016)

4.4.1 Pantalla de ingreso:

package Presentation [Presentation Index]

Index

:imagen inicial



:texto Sistema para reserva de parqueaderos PUCE ≈

: Usuario



: Clave



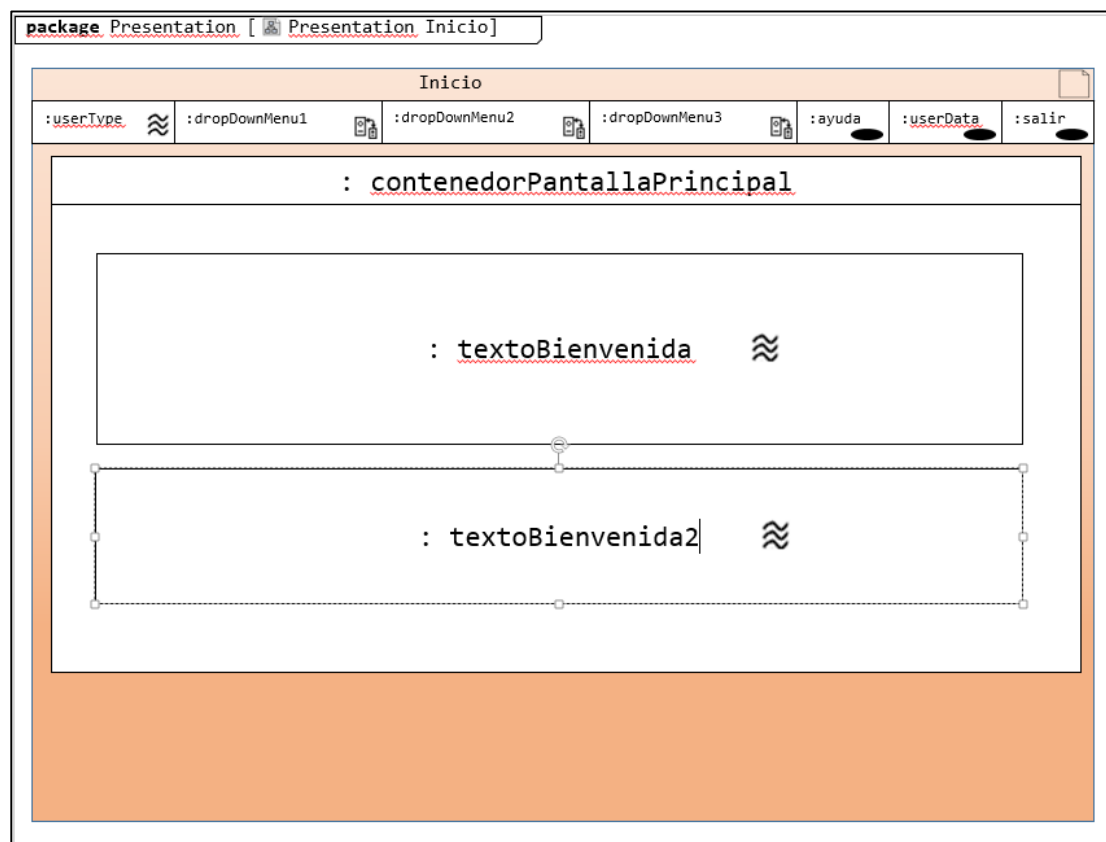
:ingresar



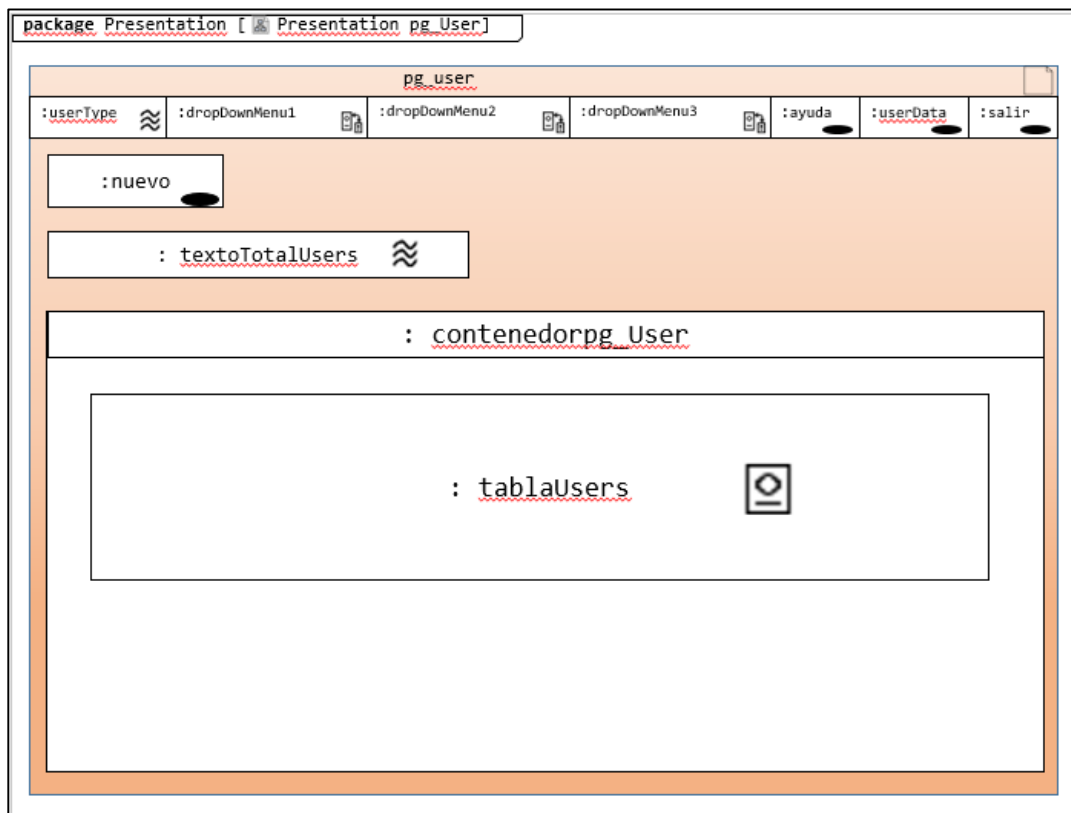
:imagen logo
PUCE



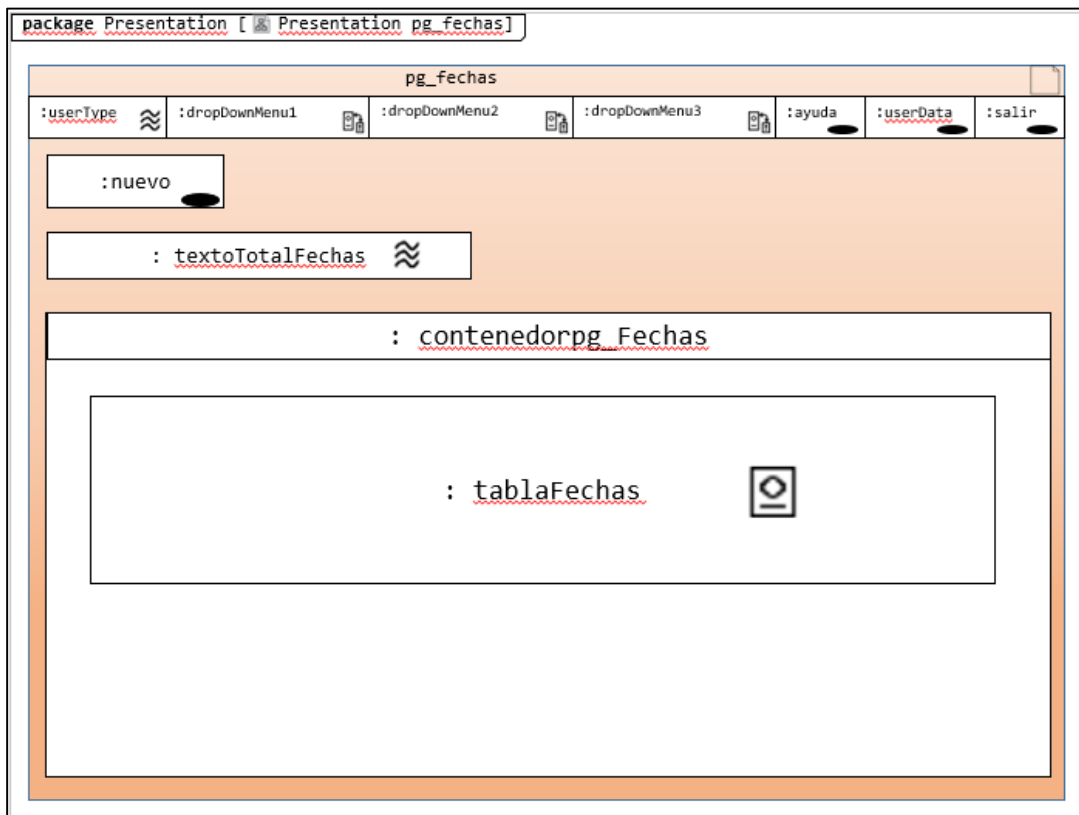
4.4.2 Pantalla principal



4.4.3 Creación de usuarios



4.4.4 Creación fechas de reservas



4.4.5 Reportes de infracciones

package Presentation [Presentation infracciones]

infracciones

:userType

:dropDownMenu1

:dropDownMenu2

:dropDownMenu3

:ayuda

:userData

:salir

: contenedor_Infracciones

: texto_seleccione fecha Inicio y final

: texto_fecha_Inicio

: fecha_inicio

: texto_fecha_final

: fecha_final

:generar reporte

:cancelar

4.4.6 Reportes de reservas

package Presentation [Presentation reservas_rep]

reservas_rep

:userType

:dropDownMenu1

:dropDownMenu2

:dropDownMenu3

:ayuda

:userData

:salir

: contenedor_reservas_rep

: texto_seleccione_sector fecha horario

: texto_fecha

: selccione_fecha

: texto_sector

: sector

: texto_horario_inicio

: horario_inicio

: texto_horario_fin

: horario_fin

:generar

:cancelar

4.4.7 Reporte de movimientos vehiculares

package Presentation [Presentation movimientos]

movimientos

:userType :dropDownMenu1 :dropDownMenu2 :dropDownMenu3 :ayuda :userData :salir

: movimientos

: texto_seleccione_fehca_y_horas

: texto fecha : selccione_fecha

: texto horario inicio

: horario_inicio

: texto horario fin

: horario_fin

:generar :cancelar

4.4.8 Editar información de usuario

package Presentation [Presentation modificarUsuario]

modificarUsuarios

:userType :dropDownMenu1 :dropDownMenu2 :dropDownMenu3 :ayuda :userData :salir

: modificarUusarios

: texto nombres

: nombres

: texto apellidos

: apellidos

: texto nombre usuario

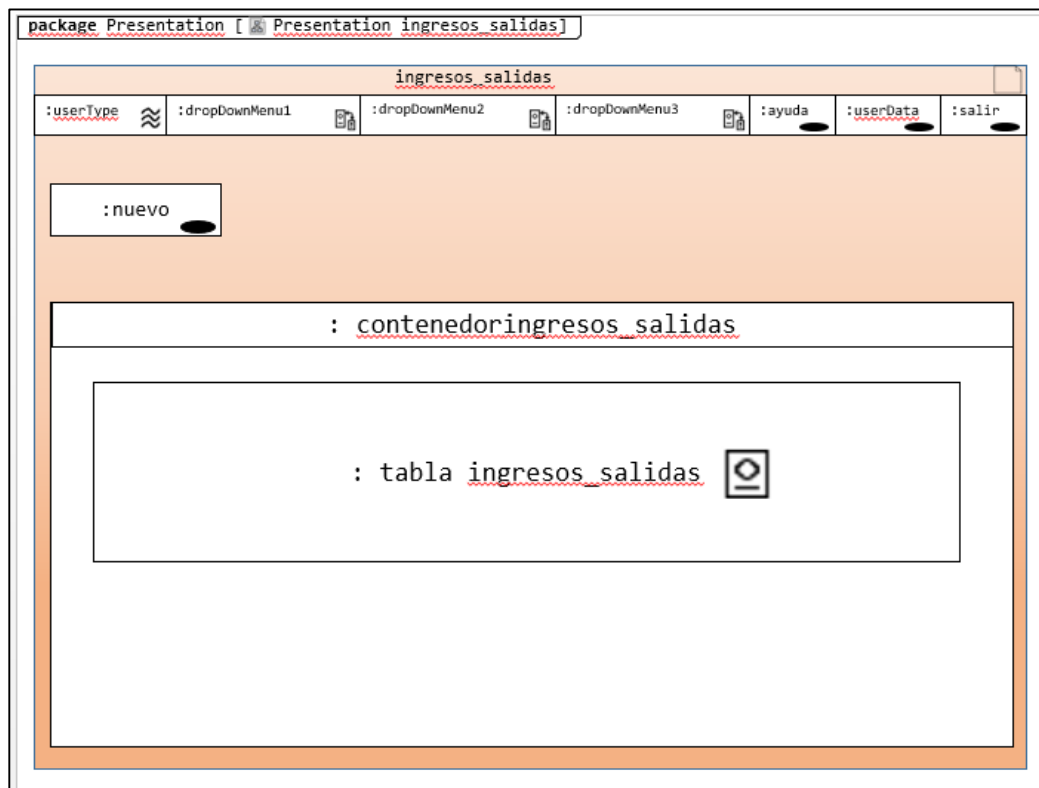
: nombre usuario

: texto clave usuario

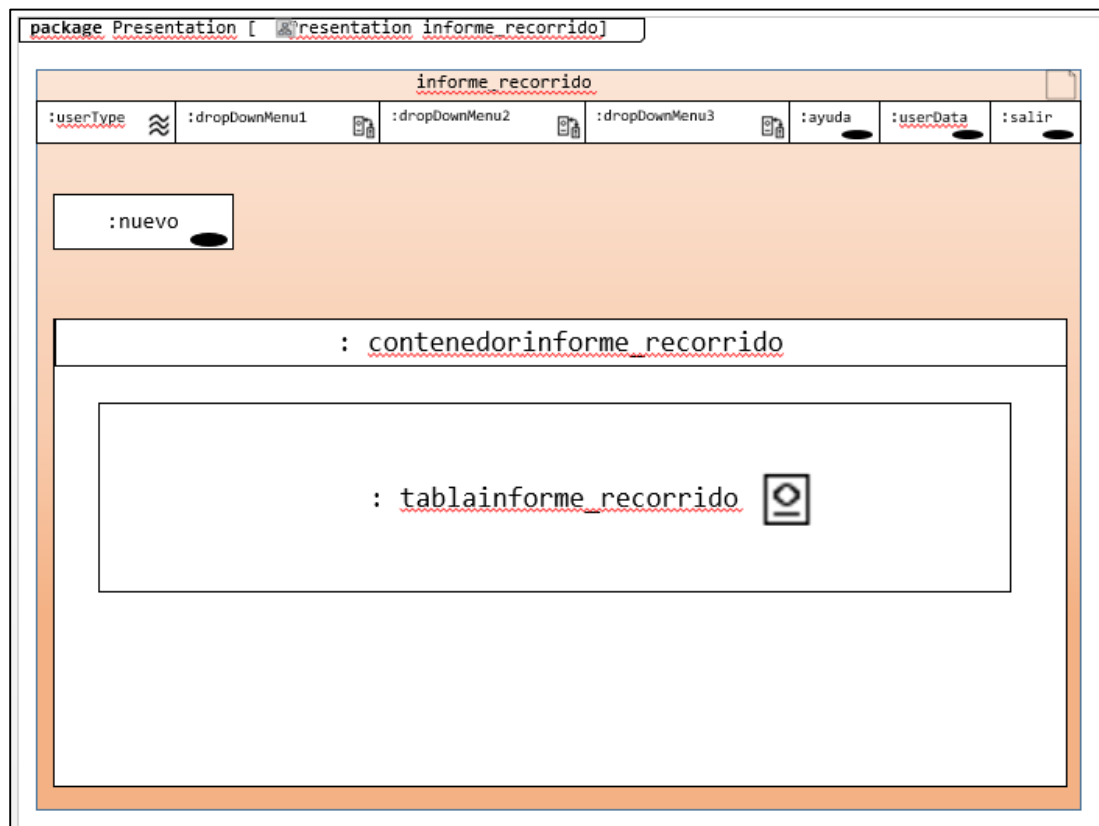
: clave usuario

:generar :cancelar

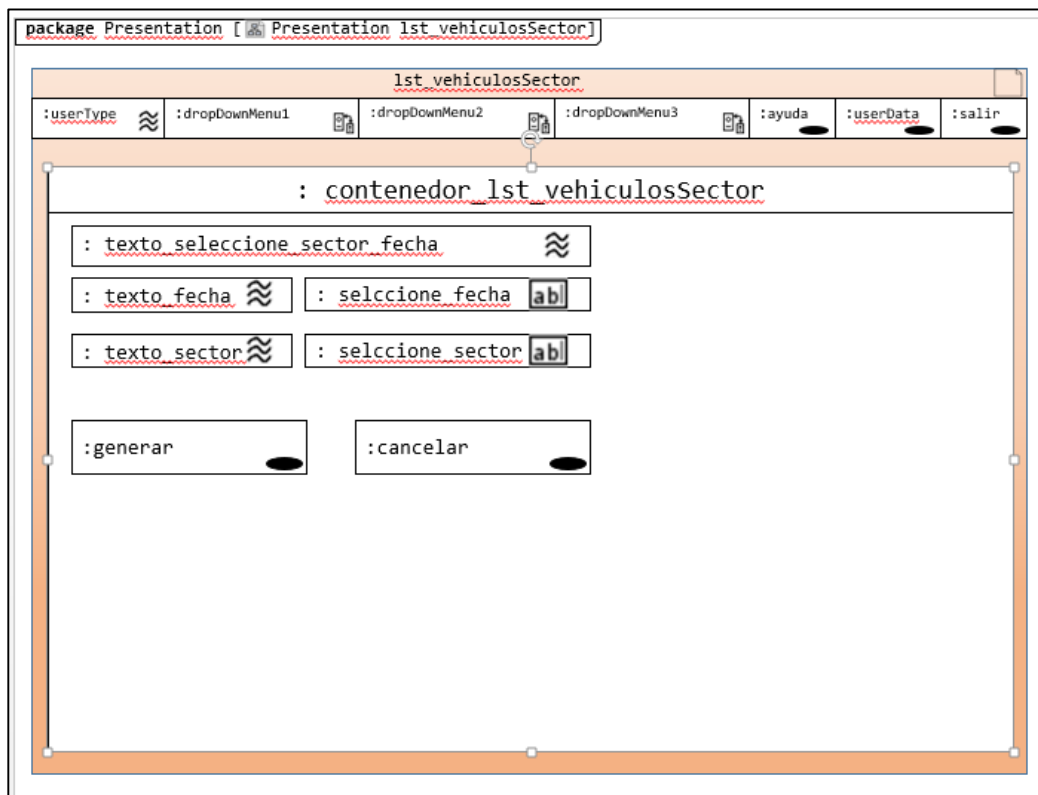
4.4.9 Registro de ingresos y salidas



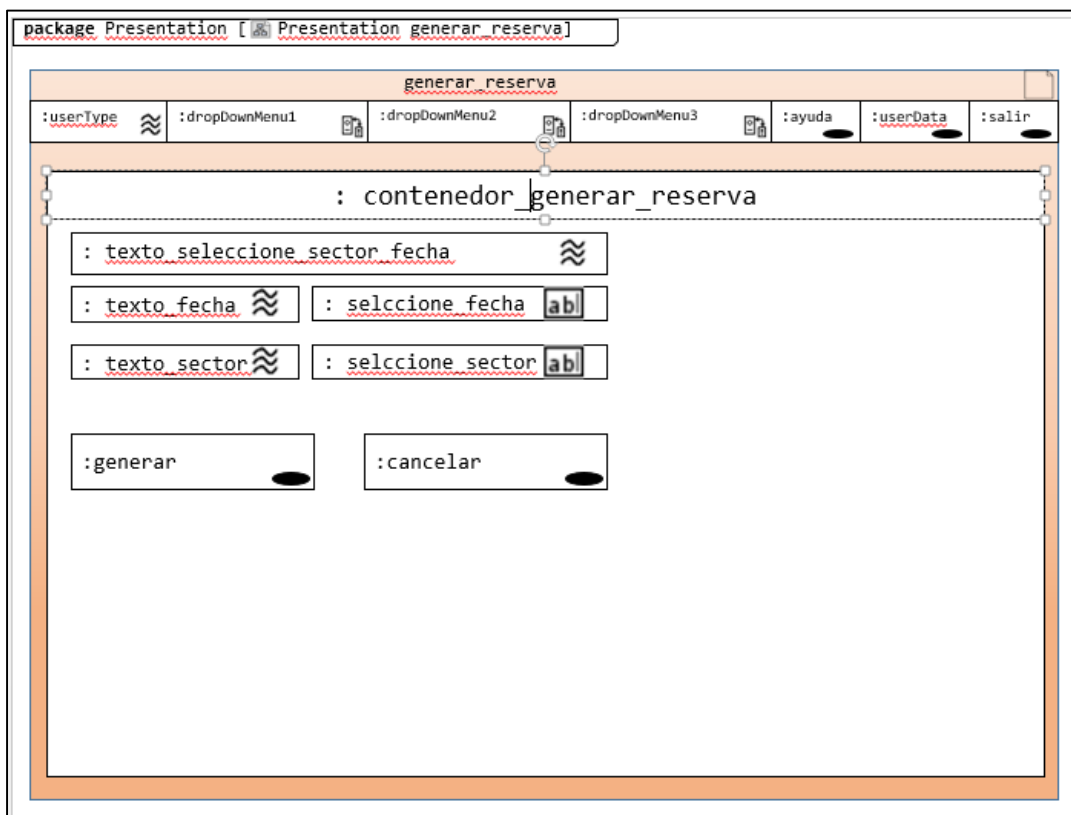
4.4.10 Informes de recorrido



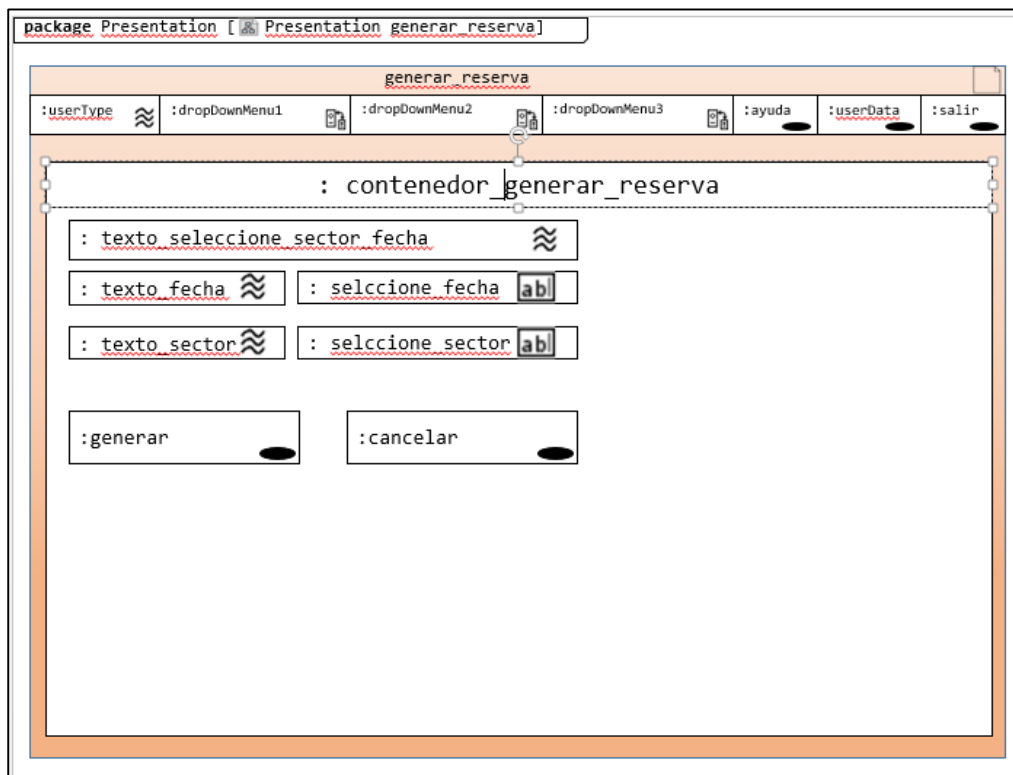
4.4.11 Generar lista de vehículos por sector



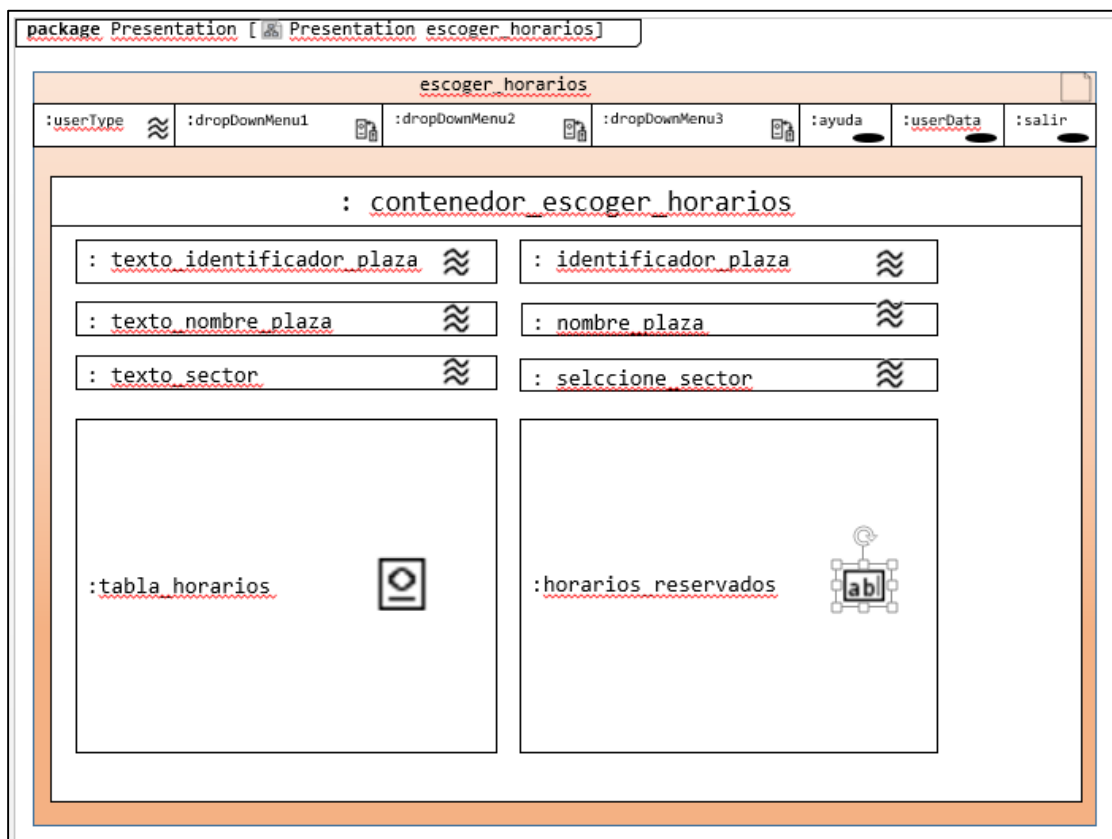
4.4.12 Generar reserva por parte del usuario



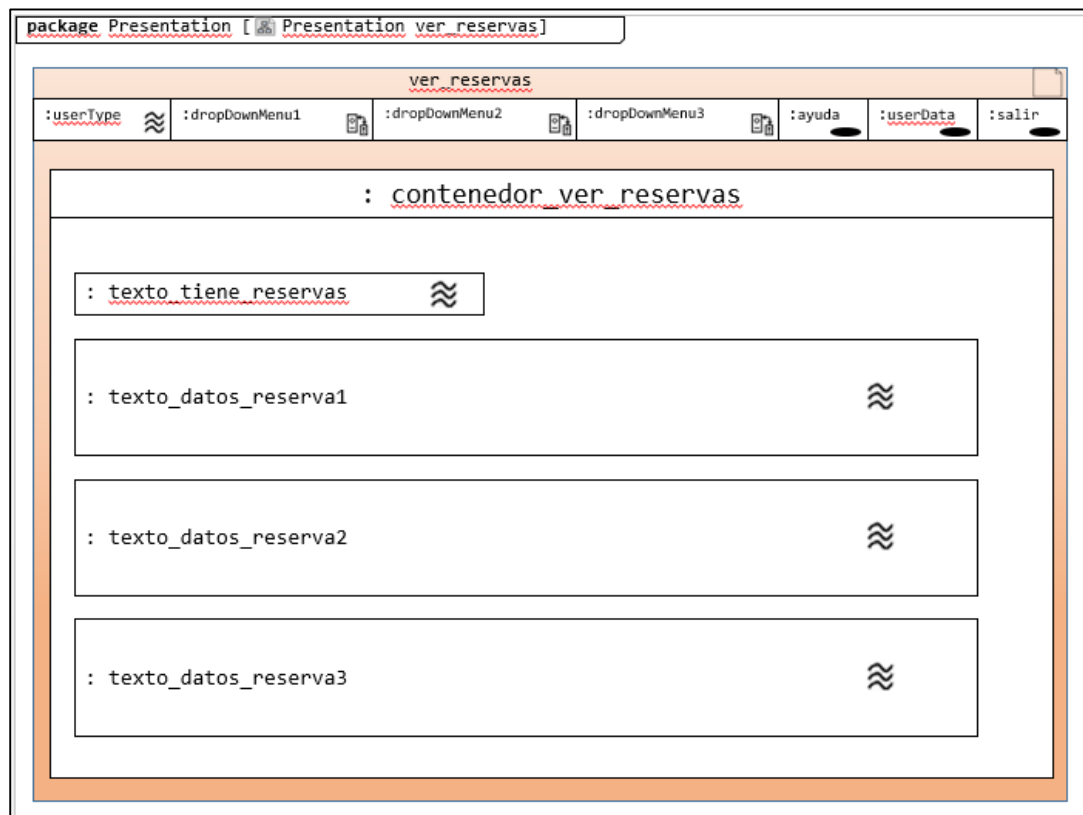
4.4.13 Seleccionar plazas



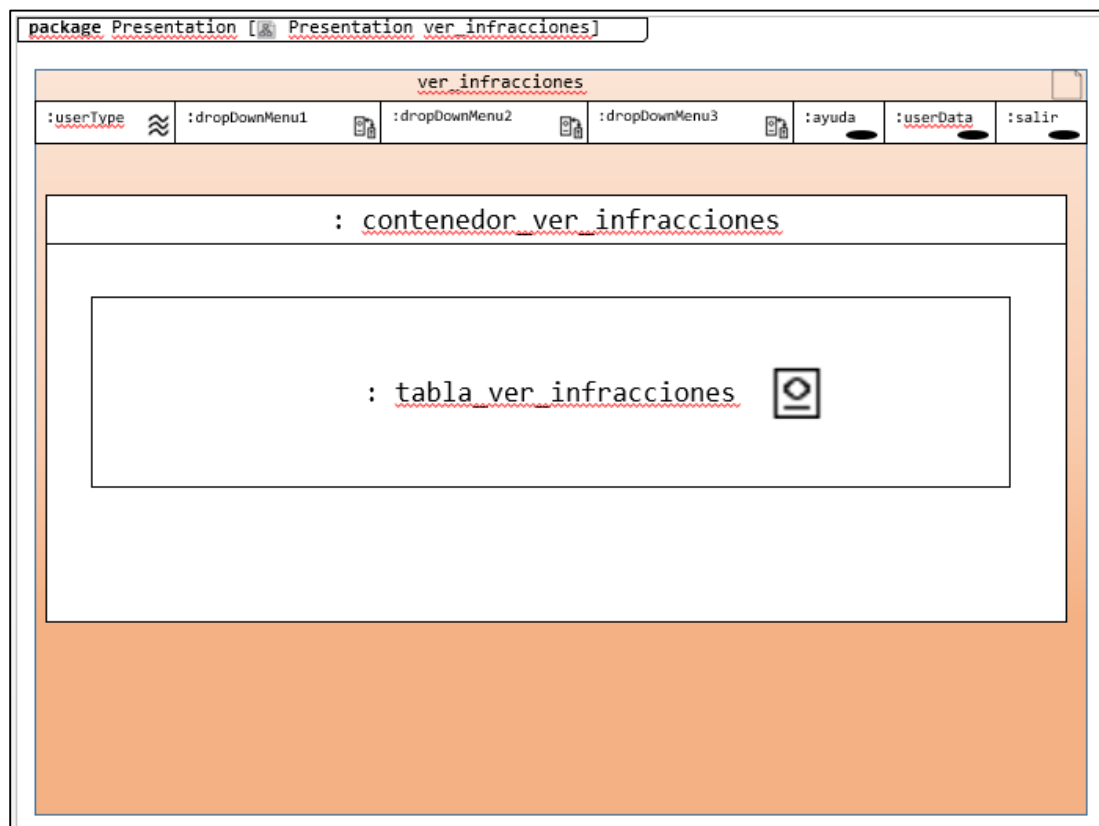
4.4.14 Escoger horarios de plazas



4.4.15 Ver reservas realizadas

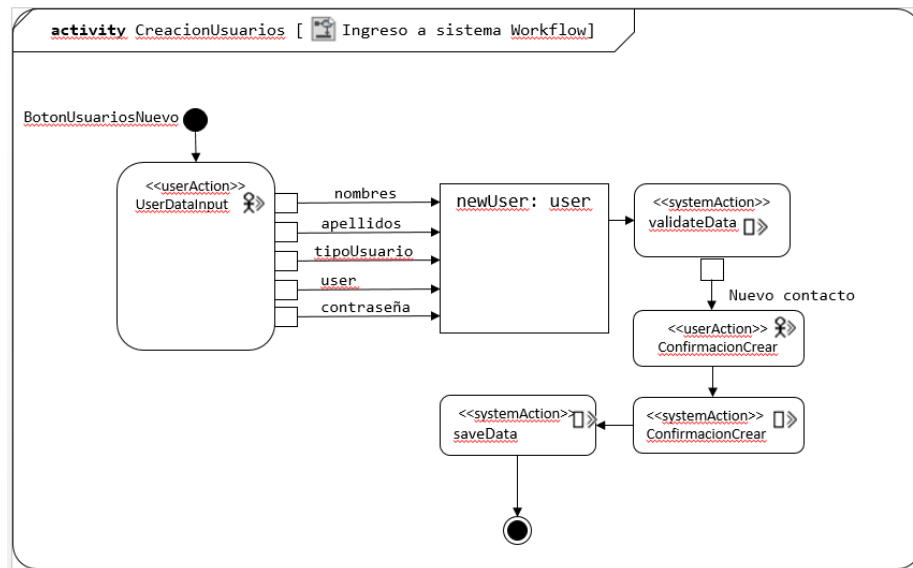


4.4.16 Ver reservas de usuarios

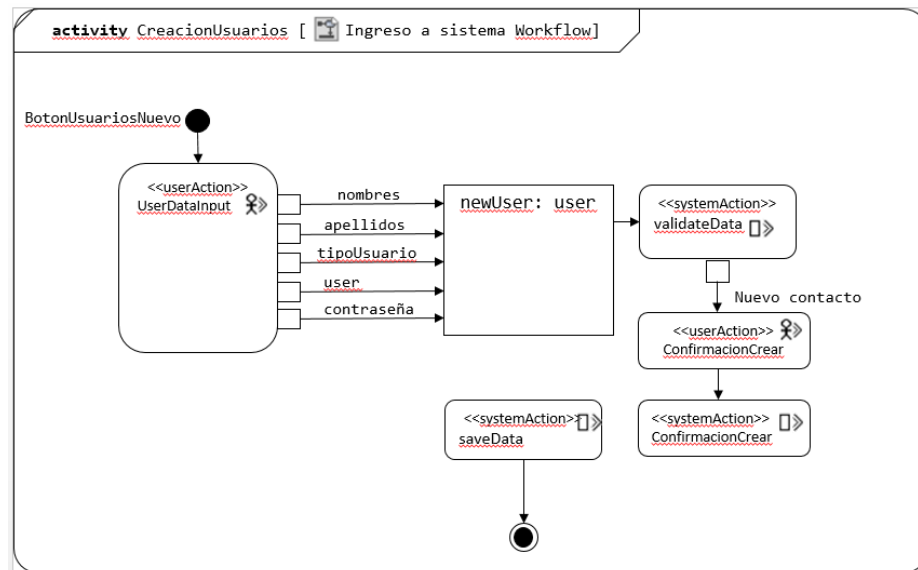


4.5 MODELO DE PROCESOS

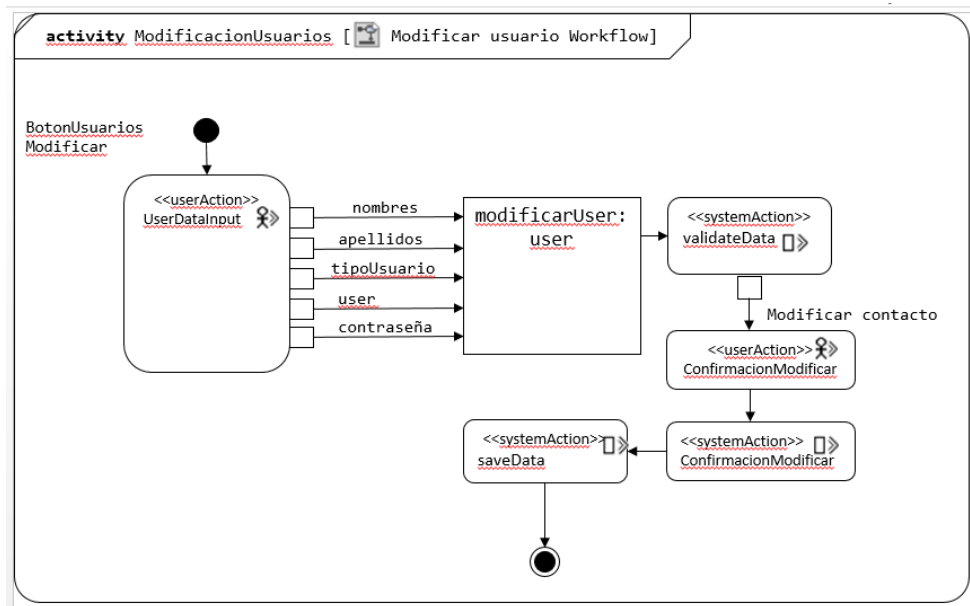
4.5.1 Ingreso al sistema



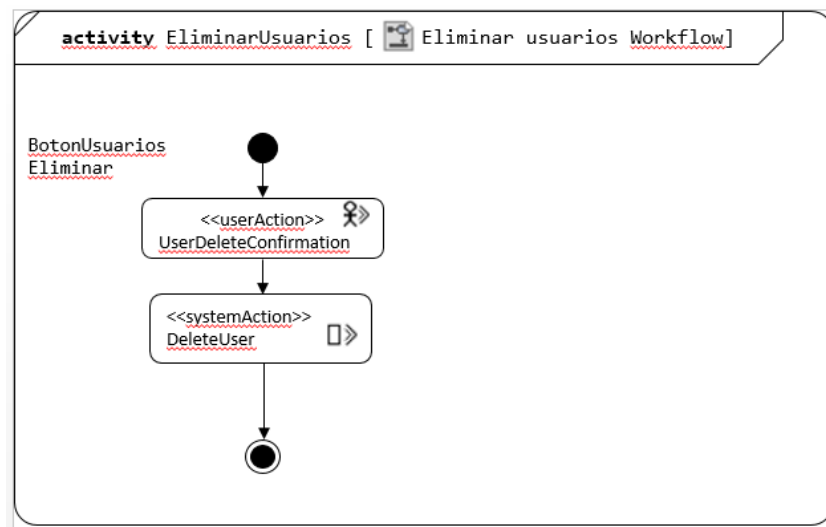
4.5.2 Creación de usuarios



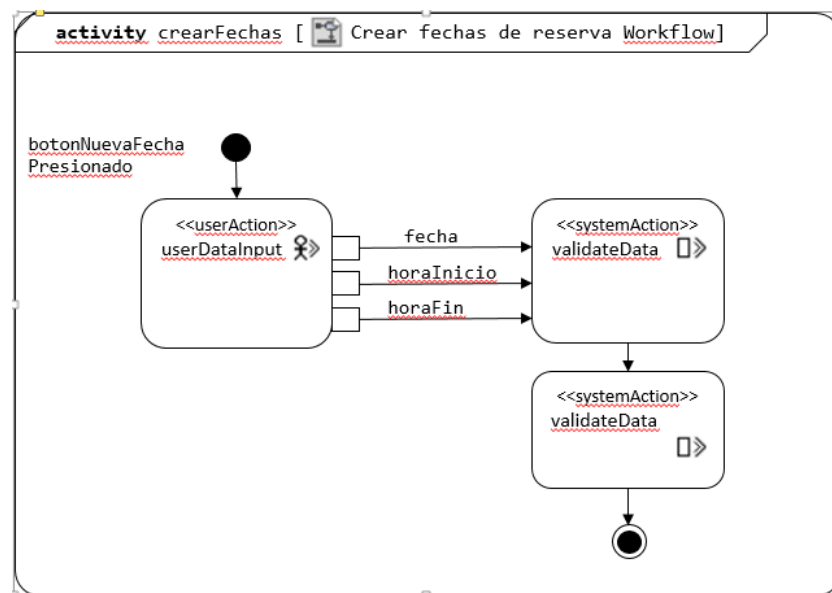
4.5.3 Modificación de usuarios



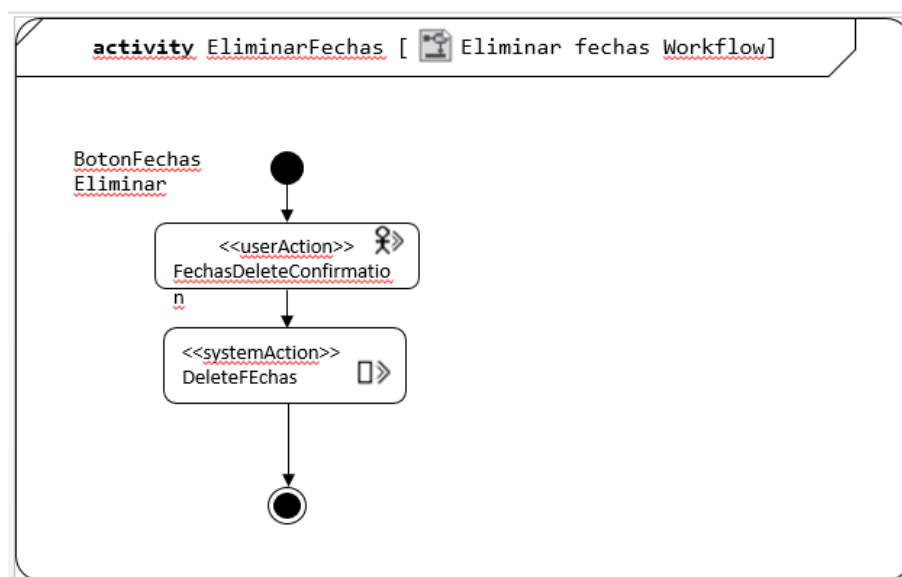
4.5.4 Eliminar usuarios



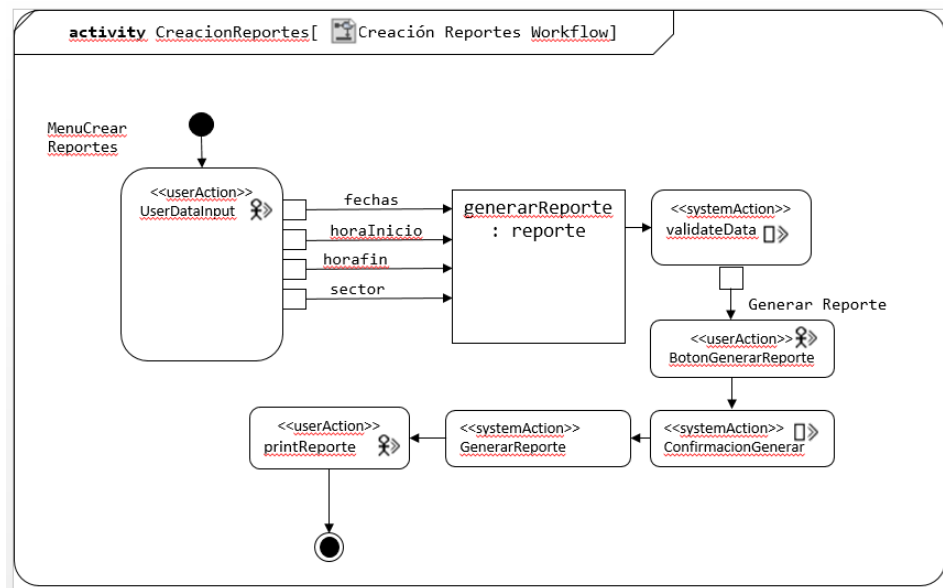
4.5.5 Crear fechas de reservas



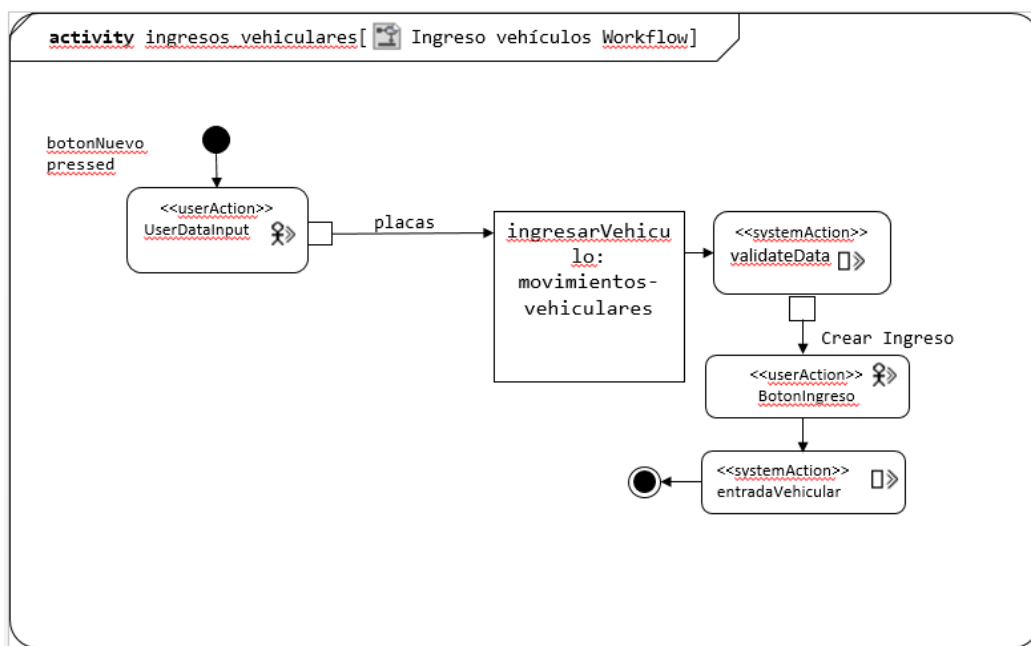
4.5.6 Eliminar fechas



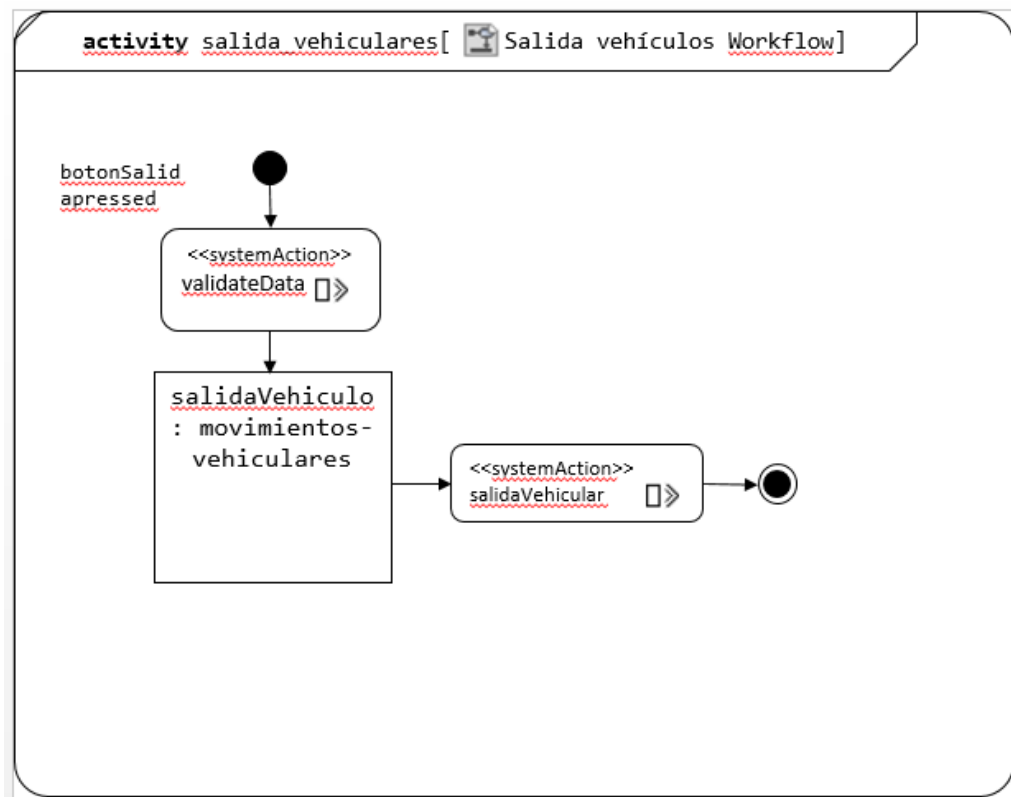
4.5.7 Generar reportes



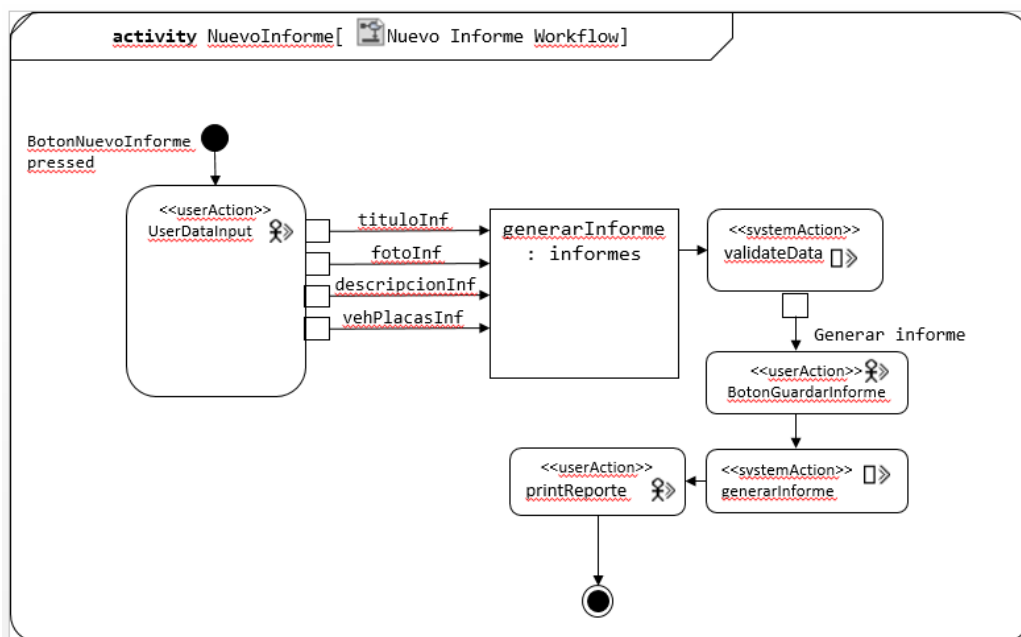
4.5.8 Registro de ingresos vehiculares



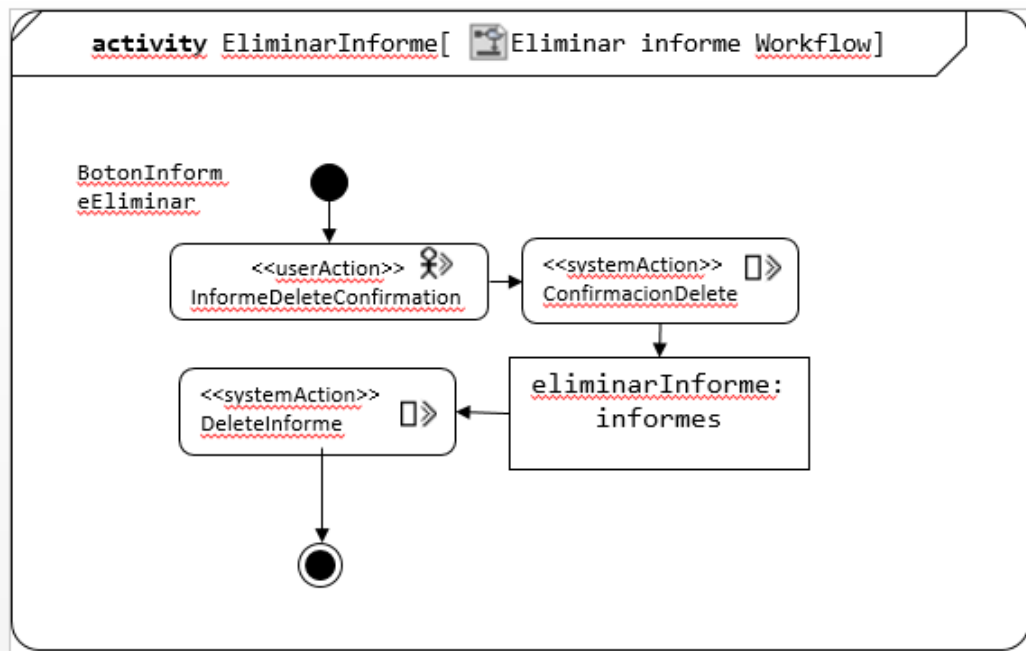
4.5.9 Registro de salidas vehiculares



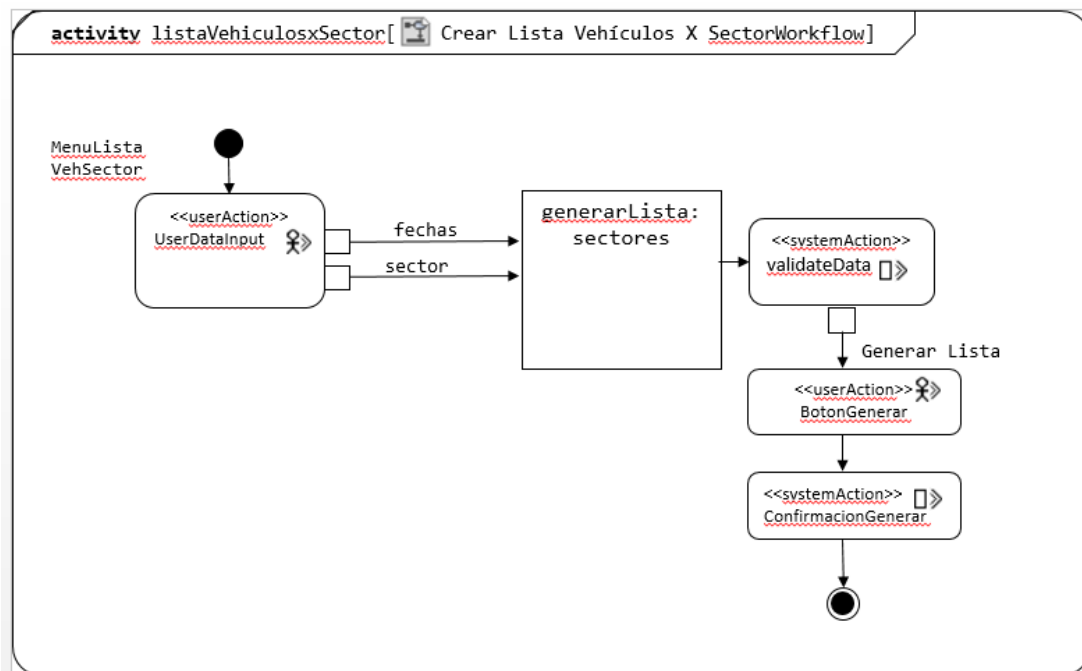
4.5.10 Nuevos informes



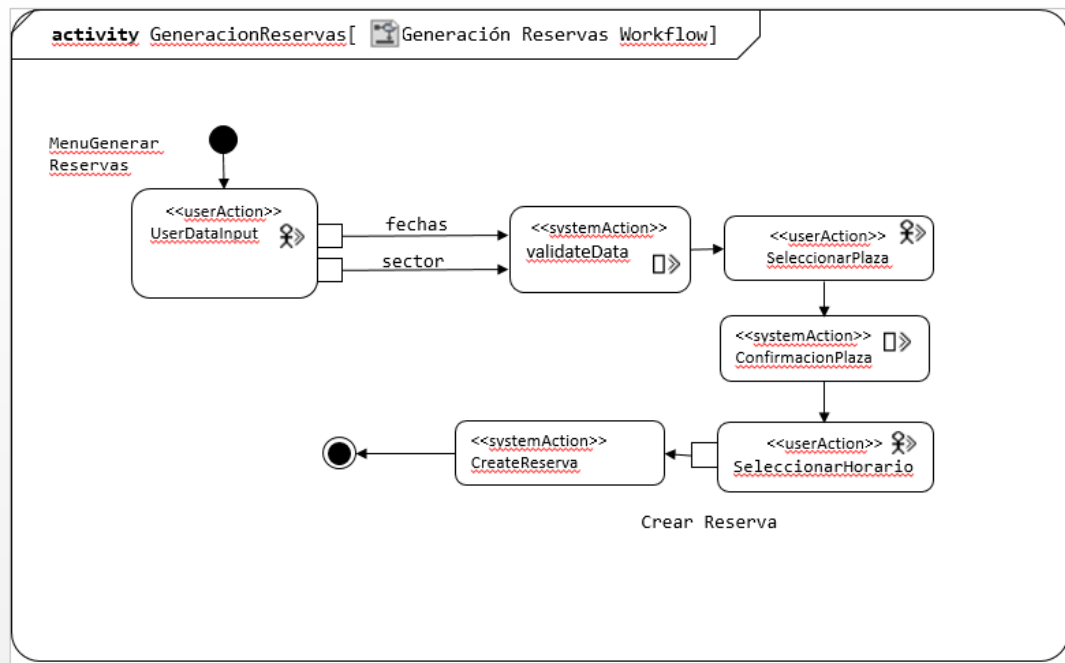
4.5.11 Eliminar informes



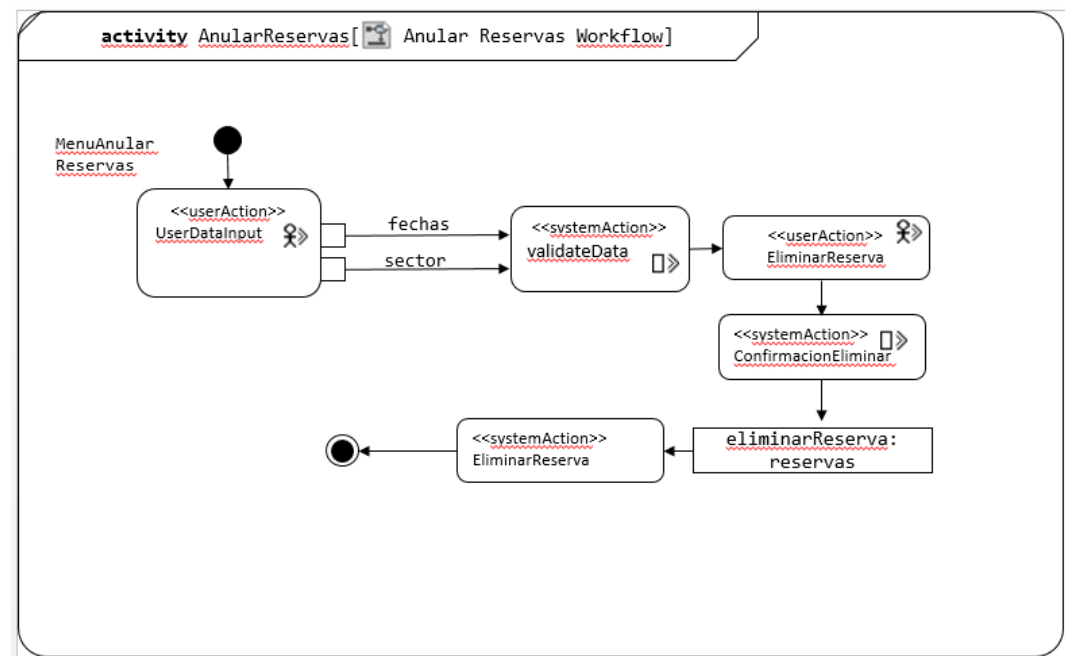
4.5.12 Generar lista de vehículos por sector



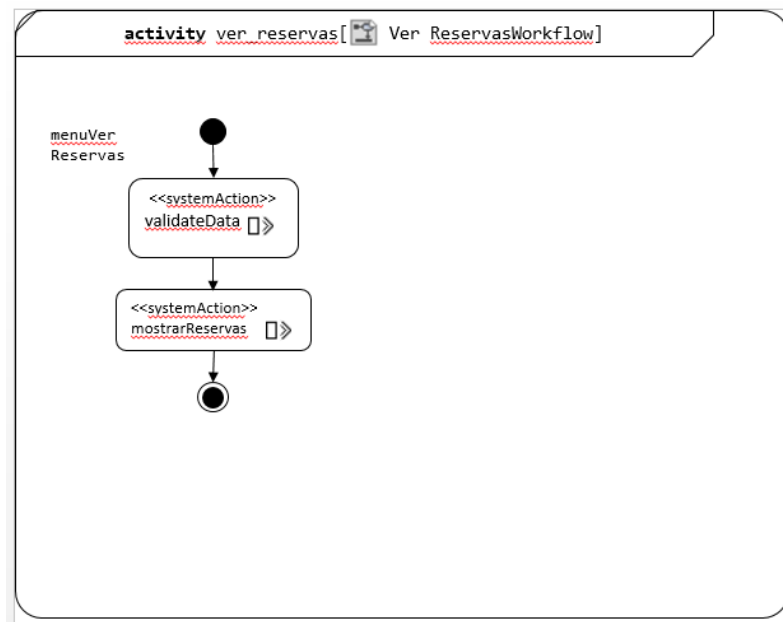
4.5.13 Generar reservas



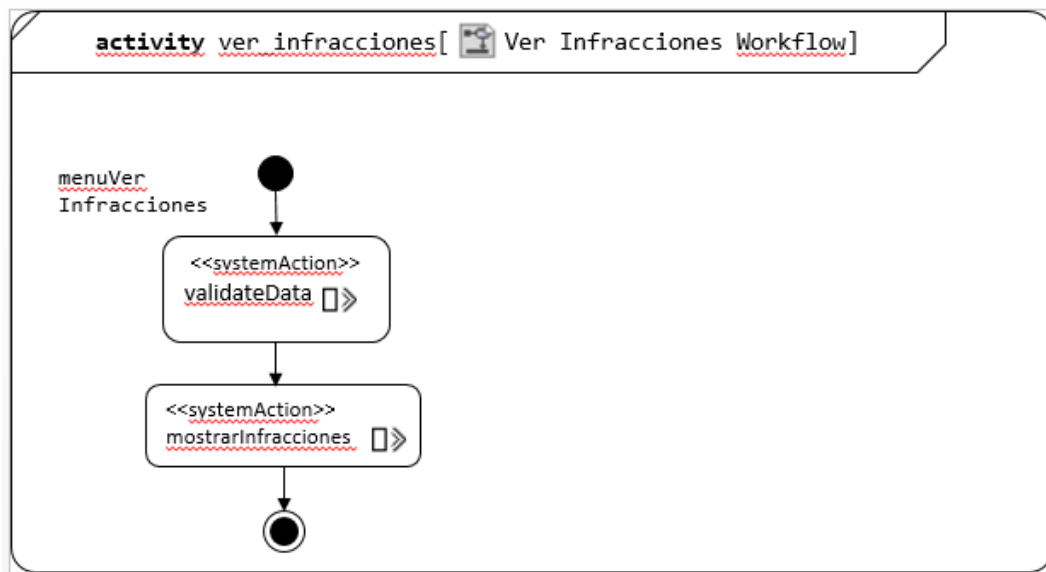
4.5.14 Anular reservas



4.5.15 Información reservas



4.5.16 Información infracciones



5 CAPÍTULO IV: DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA WEB

5.1 CONFIGURACIÓN DE HERRAMIENTAS

5.1.1 Servidor de base de datos

Servidor: 127.0.0.1 via TCP/IP

Tipo de servidor: MariaDB

Versión del servidor: 10.1.10-MariaDB - mariadb.org binary distribution

Versión del protocolo: 10

Usuario: root@localhost

Conjunto de caracteres del servidor: UTF-8 Unicode (utf8)

5.1.2 Servidor web

Apache/2.4.17 (Win32) OpenSSL/1.0.2d PHP/5.6.15

Versión del cliente de base de datos: libmysql - mysqlnd 5.0.11-dev - 20120503 - \$Id: 3c688b6bbc30d36af3ac34fdd4b7b5b787fe5555 \$

Extensión PHP: mysqli Documentación

Versión de PHP: 5.6.15

5.1.3 phpMyAdmin

Versión utilizada: 4.5.1, versión estable más reciente: 4.6.4

5.2 HERRAMIENTAS DEL SISTEMA

5.2.1 Librerías

5.2.1.1 JQUERY

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
```

5.2.1.2 BOOTSTRAP

```
<script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
```

5.2.1.3 BOOTSTRAP SELECT

```
<script src="librerias/dist/js/bootstrap-select.js"></script>
```

5.2.2 Hojas de estilo

5.2.2.1 BOOTSTRAP

```
<link rel="stylesheet"
```

```
href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
```

5.2.2.2 BOOTSTRAP SELECT

```
<link rel="stylesheet" href="librerias/dist/css/bootstrap-select.css">
```

5.3 CÓDIGO DE DESARROLLO

5.3.1 Conexión a la base de datos

```
<?php
```

```
# FileName="Connection_php_mysql.htm"
```

```
# Type="MYSQL"
```

```
# HTTP="true"
```

```
$hostname_conexion = "localhost";
```

```
$database_conexion = "bdd_parqueaderos";
```

```
$username_conexion = "root";
```

```
$password_conexion = "1722942941";
```

```
$conexion = mysql_pconnect($hostname_conexion, $username_conexion,  
$password_conexion) or trigger_error(mysql_error(),E_USER_ERROR);
```

```
?>
```

5.3.2 Manejo de funciones por versión PHP

```
<?php
```

```
if (!function_exists("GetSQLValueString")) {
```

```
function GetSQLValueString($theValue, $theType, $theDefinedValue = "",  
$theNotDefinedValue = "")
```

```
{
```

```
if (PHP_VERSION < 6) {
```

```
    $theValue = get_magic_quotes_gpc() ? stripslashes($theValue) : $theValue;
```

```
}
```

```
$theValue = function_exists("mysql_real_escape_string") ?
```

```
mysql_real_escape_string($theValue) : mysql_escape_string($theValue);
```

```
switch ($theType) {
```

```
    case "text":
```

```
$theValue = ($theValue != "") ? "" . $theValue . "" : "NULL";
```

```
break;
```

```
case "long":
```

```
case "int":
```

```
$theValue = ($theValue != "") ? intval($theValue) : "NULL";
```

```
break;
```

```
case "double":
```

```
$theValue = ($theValue != "") ? doubleval($theValue) : "NULL";
```

```
break;
```

```
case "date":
```

```
$theValue = ($theValue != "") ? "" . $theValue . "" : "NULL";
```

```
break;
```

```
case "defined":
```

```
$theValue = ($theValue != "") ? $theDefinedValue : $theNotDefinedValue;
```

```
break;
```

```
}
```

```
return $theValue;
```

```
}
```

```
}
```

```
?>
```

5.3.3 Inicios de sesión

```
<?php

// *** Validate request to login to this site.

if (!isset($_SESSION)) {

    session_start();

}

$loginFormAction = $_SERVER['PHP_SELF'];

if (isset($_GET['accesscheck'])) {

    $_SESSION['PrevUrl'] = $_GET['accesscheck'];

}

if (isset($_POST['usr'])) {

    $loginUsername=$_POST['usr'];

    $password=$_POST['pwd'];

    $MM_fldUserAuthorization = "idTipoUsuarios";

    $MM_redirectLoginSuccess = "inicio.php";

    $MM_redirectLoginFailed = "error.php";

    $MM_redirecttoReferrer = false;

    mysql_select_db($database_conexion, $conexion);
```

```
$LoginRS__query=sprintf("SELECT  usuarioAccesoUsuario,  claveAccesoUsuario,
idTipoUsuarios  FROM  usuarios  WHERE  usuarioAccesoUsuario=%s  AND
claveAccesoUsuario=%s",
```

```
GetSQLValueString($loginUsername, "text"), GetSQLValueString($password, "text"));
```

```
$LoginRS = mysql_query($LoginRS__query, $conexion) or die(mysql_error());
```

```
$loginFoundUser = mysql_num_rows($LoginRS);
```

```
if ($loginFoundUser) {
```

```
    $loginStrGroup = mysql_result($LoginRS,0,'idTipoUsuarios');
```

```
        if (PHP_VERSION >= 5.1) {session_regenerate_id(true);} else
{session_regenerate_id();}
```

```
//declare two session variables and assign them
```

```
$_SESSION['MM_Username'] = $loginUsername;
```

```
$_SESSION['MM_UserGroup'] = $loginStrGroup;
```

```
if (isset($_SESSION['PrevUrl']) && false) {
```

```
    $MM_redirectLoginSuccess = $_SESSION['PrevUrl'];
```

```
}
```

```
header("Location: " . $MM_redirectLoginSuccess );
```

```
}
```

```
else {
```

```
header("Location: ". $MM_redirectLoginFailed );

}

}

?>
```

5.3.4 Manejo de Menús a presentar por tipo de usuario

```
<?php

switch ($_SESSION['MM_UserGroup']-1) {

    case 0:

        require_once('menuUsuario.php');

        break;

    case 1:

        require_once('menuMaster.php');

        break;

    case 2:

        require_once('menuAsistente.php');

        break;

    case 3:

        require_once('menuControl.php');
```

```
        break;

    case 4:

        require_once('menuRecorrido.php');

        break;

    }

?>
```

5.3.5 Manejo de ficheros

```
<?php

if( !isset($_FILES['archivo']) ){

    echo 'Ha habido un error, tienes que elegir un archivo<br/>';

}else{

    $nombre = $_FILES['archivo']['name'];

    $nombre_tmp = $_FILES['archivo']['tmp_name'];

    $tipo = $_FILES['archivo']['type'];

    $tamano = $_FILES['archivo']['size'];

    echo 'Nombre: ' . $nombre . '<br/>';

    echo 'Tipo: ' . $tipo . '<br/>';

    echo 'Tamaño: ' . ($tamano / 1024) . ' Kb<br/>';
```

```

echo 'Guardado en: ' . $nombre_tmp;

if( file_exists( 'subidas/'.$nombre) )

{

    echo '<br/>El archivo ya existe: ' . $nombre;

}

else

{

    move_uploaded_file($nombre_tmp,

        "subidas/" . $nombre);

    echo "<br/>Guardado en: " . "subidas/" . $nombre;

}

}

$insertSQL = "INSERT INTO informes (tipoInforme, fotoInforme, descripcionInforme,
nombreUsuario,          idVehiculo,fechaInforme,          horaInforme)          VALUES
('".$_POST['tipoInforme']."','".$nombre."','".$_POST['descripcionInforme']."','".$_POST['no
mbreUsuario']."','".$_POST['idVehiculo']."',".date("Y/m/d")."',".date("G:H:s")."');

echo $insertSQL;

mysql_select_db($database_conexion, $conexion);

$result1 = mysql_query($insertSQL, $conexion) or die(mysql_error());

```



```
$insertGoTo = "informe_recorrido.php";

header(sprintf("Location: %s", $insertGoTo));

?>
```

5.3.6 Consultas básicas a la base de datos

```
mysql_select_db($database_conexion, $conexion);

$query_informes = sprintf("SELECT * FROM informes WHERE idInforme = %s",
GetSQLValueString($colname_informes, "int"));

$informes = mysql_query($query_informes, $conexion) or die(mysql_error());

$row_informes = mysql_fetch_assoc($informes);

$totalRows_informes = mysql_num_rows($informes);
```

5.3.7 Restricción de páginas

```
<?php

//initialize the session

if (!isset($_SESSION)) {

    session_start();
```

```
}
```

```
// ** Logout the current user. **
```

```
$logoutAction = $_SERVER['PHP_SELF']."?doLogout=true";
```

```
if ((isset($_SERVER['QUERY_STRING']) && ($_SERVER['QUERY_STRING'] != "")){
```

```
    $logoutAction .= "&". htmlentities($_SERVER['QUERY_STRING']);
```

```
}
```

```
if ((isset($_GET['doLogout'])) && ($_GET['doLogout']=="true")){
```

```
    //to fully log out a visitor we need to clear the session variables
```

```
    $_SESSION['MM_Username'] = NULL;
```

```
    $_SESSION['MM_UserGroup'] = NULL;
```

```
    $_SESSION['PrevUrl'] = NULL;
```

```
    unset($_SESSION['MM_Username']);
```

```
    unset($_SESSION['MM_UserGroup']);
```

```
    unset($_SESSION['PrevUrl']);
```

```
$logoutGoTo = "index.php";
```

```
if ($logoutGoTo) {
```

```
    header("Location: $logoutGoTo");
```

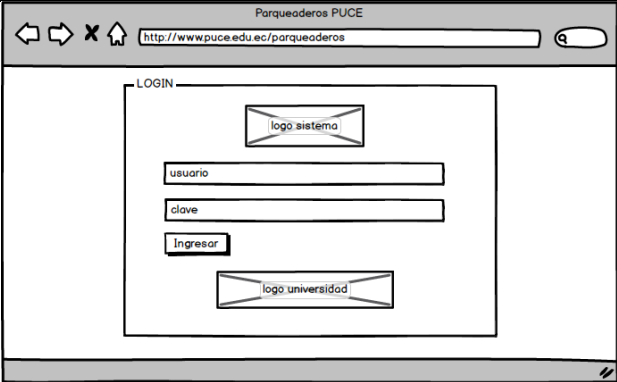
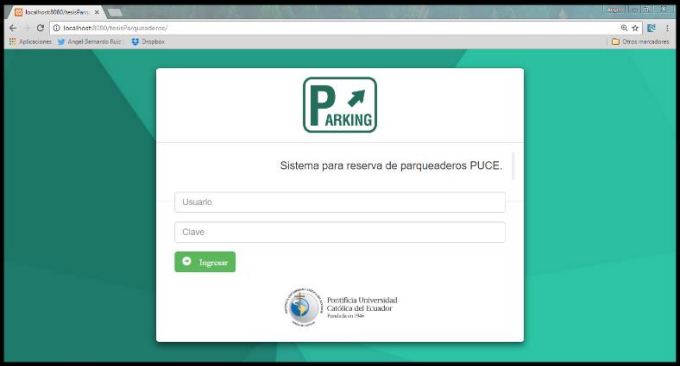
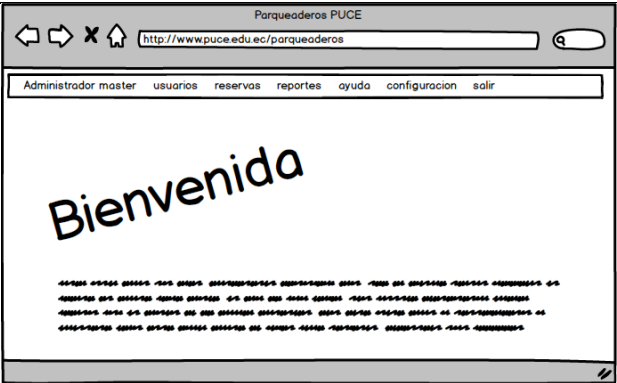

exit;

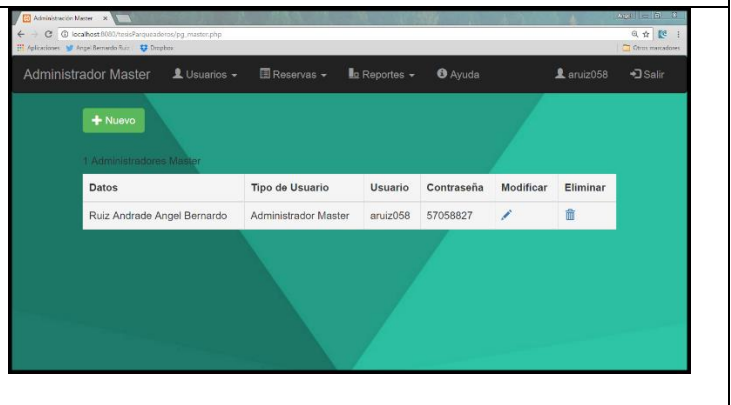
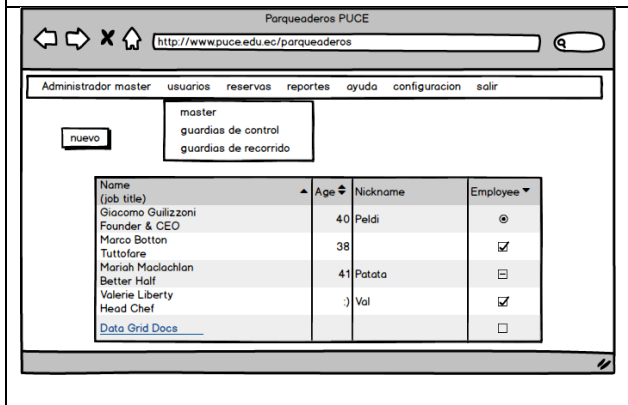
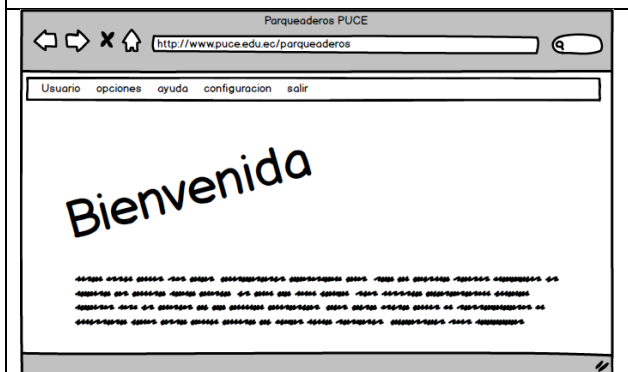
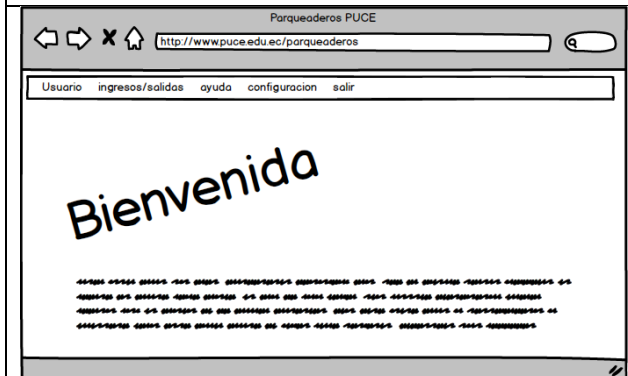
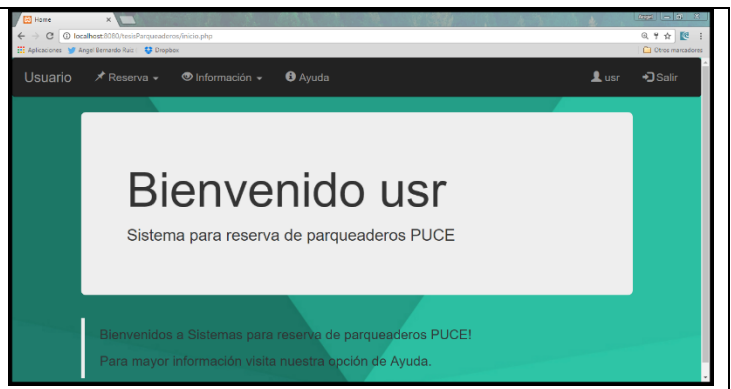
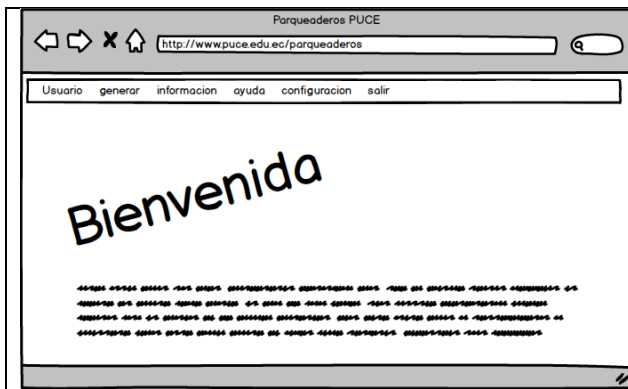
}

}

?>

5.4 PROTOTIPADO Y DISEÑO

Prototipo	Construcción final
	
	



Parqueaderos PUCE

Administrador master usuarios reservas reportes ayuda configuracion salir

master
guardias de control
guardias de recorrido

Guardar

Administrador Master

Usuarios Reservas Reportes Ayuda

Nombre:

Apellido:

Tipo de Usuario:
Administrador Master

Nombre de Usuario:

Clave de Usuario:

Guardar Cancelar

Parqueaderos PUCE

Administrador master usuarios reservas reportes ayuda configuracion salir

nuevo generar fecha

nuevo

Guardar

Name (job title)	Age	Nickname	Employee
Giacomo Guizzoni Founder & CEO	40	Peldi	

Administrador Master

Usuarios Reservas Reportes Ayuda

+ Nuevo

Fecha:
00/mm/aaaa

Inicio:
07:00:00

Fin:
07:00:00

Generar Cancelar

Fecha	Inicio	Final	Eliminar
2019-10-05	07:00:00	22:00:00	

Parqueaderos PUCE

Administrador master usuarios reservas reportes ayuda configuracion salir

reporte 1
reporte 2
reporte 3
reporte 4

reporte

generar

Administrador Master

Usuarios Reservas Reportes Ayuda

Selecciona el vector, la fecha y los horarios.

Fecha:
Selecciona la fecha

Inicio:
2019-10-05

Horario Inicio:
07:00:00

Horario Fin:
07:00:00

Generar Cancelar

Parqueaderos PUCE

Administrador master usuarios reservas reportes ayuda configuracion salir

configuracion

Guardar

Modificar Usuario

Usuario Reservas Informacion Ayuda

Nombre:

Apellido:

Nombre de Usuario:

Clave de usuario:

Guardar Cancelar

Parquederos PUCE

<http://www.puce.edu.ec/parquederos>

Usuario generar informacion ayuda configuracion salir

generar reserva

sector plaza hora hora hora hora hora ☐ seleccion

plaza hora hora hora hora hora hora ☒ seleccion

plaza hora hora hora hora hora hora ☐ seleccion

plaza hora hora hora hora hora hora ☐ seleccion

plaza hora hora hora hora hora hora ☐ seleccion

plaza hora hora hora hora hora hora ☐ seleccion

plaza hora hora hora hora hora hora ☐ seleccion

plaza

Parqueaderos PUCe

http://www.puce.edu.ec/parqueaderos

Usuario generar informacion ayuda configuracion salir

Parqueaderos

placa ▼

placa ▼

placa ▼

placa ▼

placa ▼

guardar cancelar

[illegible]

Identificador Plaza:

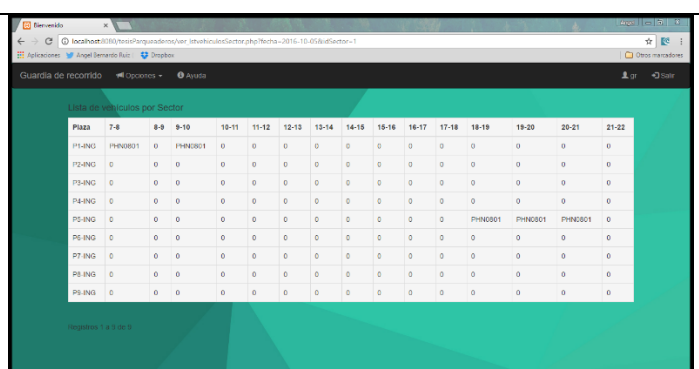
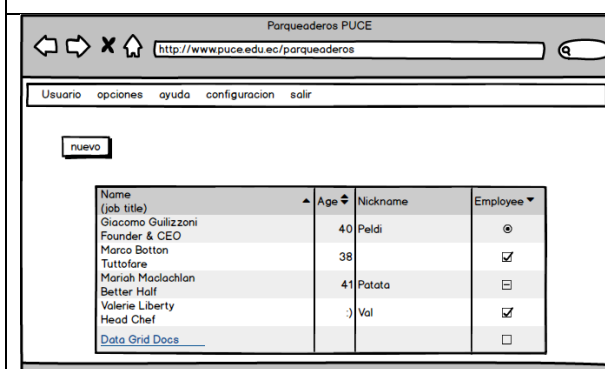
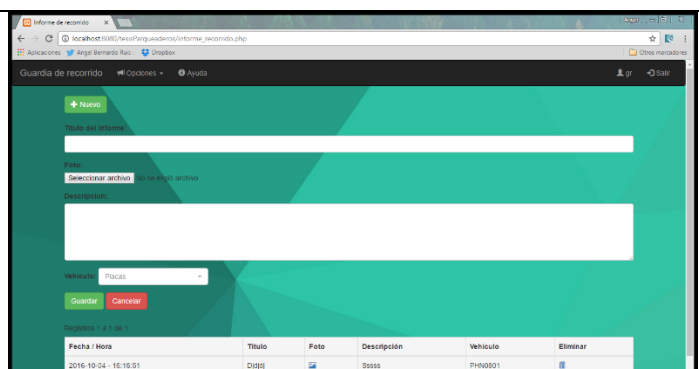
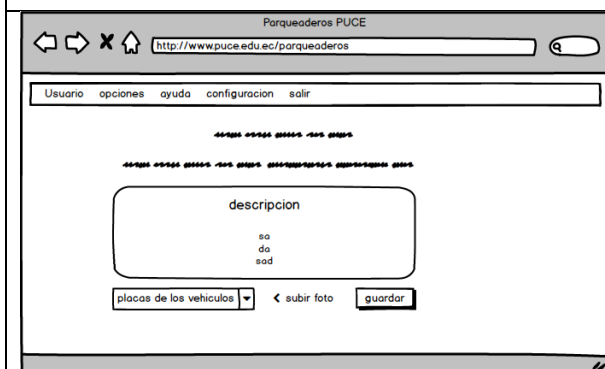
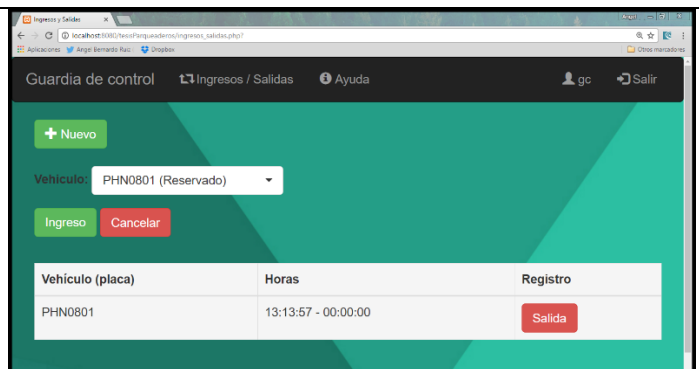
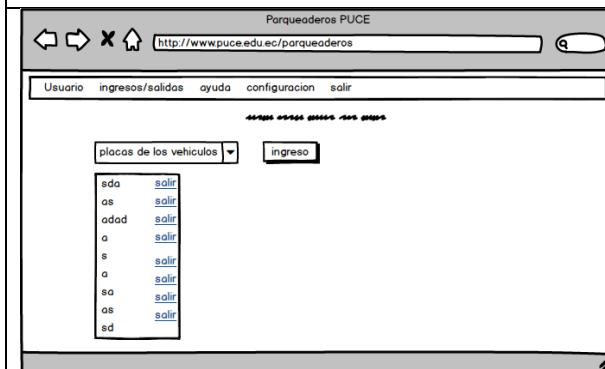
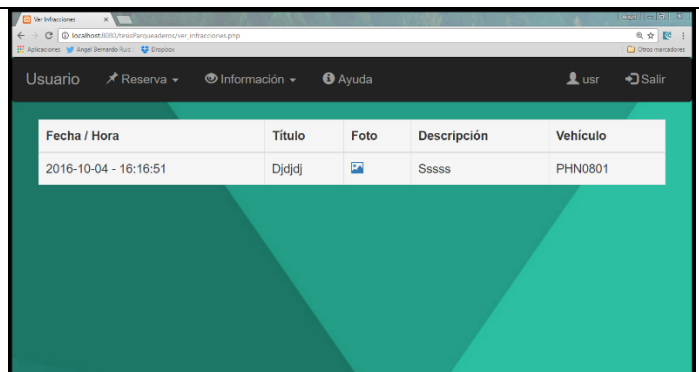
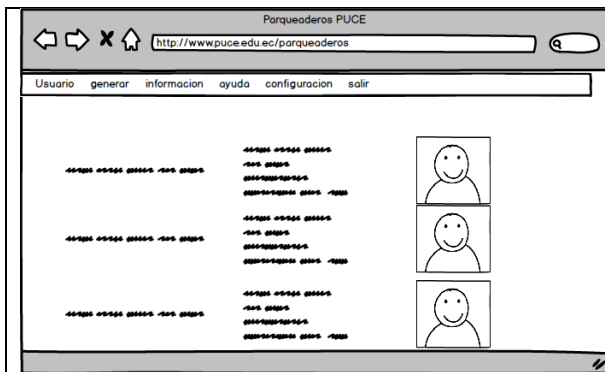
Nombre de la plaza:

Sector:

7:00-8:00	PreReservado
8:00-9:00	No reservado
9:00-10:00	PreReservado
10:00-11:00	No reservado
11:00-12:00	No reservado
12:00-13:00	No reservado
13:00-14:00	No reservado
14:00-15:00	No reservado
15:00-16:00	No reservado

The screenshot shows a web application titled "Ver Reservas". The browser's address bar indicates the URL is "localhost:8080/area/ingapoderosa/ver_reserva.php". The navigation menu includes "Usuario", "Reserva", "Información", and "Ayuda". The main heading is "Muestra todas las siguientes reservas :". Below this, there are two reservation cards:

- Card 1:**
 - Fecha : 2016-10-05
 - Pieza : P1 ING
 - Reservas : 77 / 9.38
- Card 2:**
 - Fecha : 2016-10-05
 - Pieza : P5 ING
 - Reservas : 10.16 / 10.21



6 CAPÍTULO V: PRUEBAS DE FUNCIONAMIENTO

Las pruebas se encuentran adjuntas en el documento de “Pruebas Funcionales”.

7 CAPÍTULO VI: CONCLUSIONES, RECOMENDACIONES Y ANEXOS

7.1 CONCLUSIONES

Concluimos después del análisis que se realizó sobre la situación actual de la universidad, que existe un mal manejo de las plazas disponibles para parquear el cual se podría resolver con la puesta en marcha del sistema.

En el análisis de mercado, la mayoría de personas encuestadas estarían dispuestas a utilizar el aplicativo de reservas para la universidad.

La distribución de las plazas de parqueo no es de forma equitativa ya que esta distribución depende de un sorteo que no siempre beneficia a las personas que más necesitan el espacio.

La metodología UWE es la más adecuada para realizar este tipo de proyectos web, ya que permite analizar de forma gráfica cada uno de los procesos existentes y que se van a implementar. La metodología ayuda al desarrollador a interpretar cada una de las fases del proyecto para su correcta y más sencilla implementación.

El diseño amigable y sencillo de la aplicación permite que cualquier usuario interactúe con el sistema de manera natural en cada uno de sus módulos y funcionalidades.

7.2 RECOMENDACIONES

Si se requiere implementar la aplicación en múltiples sedes, se recomienda implementar una capa donde cada una tenga la información de cada sector perteneciente a la sede.

Poner en marcha el sistema y así alimentar la base de datos para validar la integración de los datos con la aplicación.

Se debería generar una vista de la tabla que contenga la información de los usuarios para ser consumida desde el sistema de reserva de parqueaderos, así los usuarios podrán acceder al sistema con las mismas credenciales que proporciona la universidad.

Se recomienda una metodología donde las herramientas sean de software libre ya que las herramientas para el diseño y desarrollo la metodología UWE, no lo son.

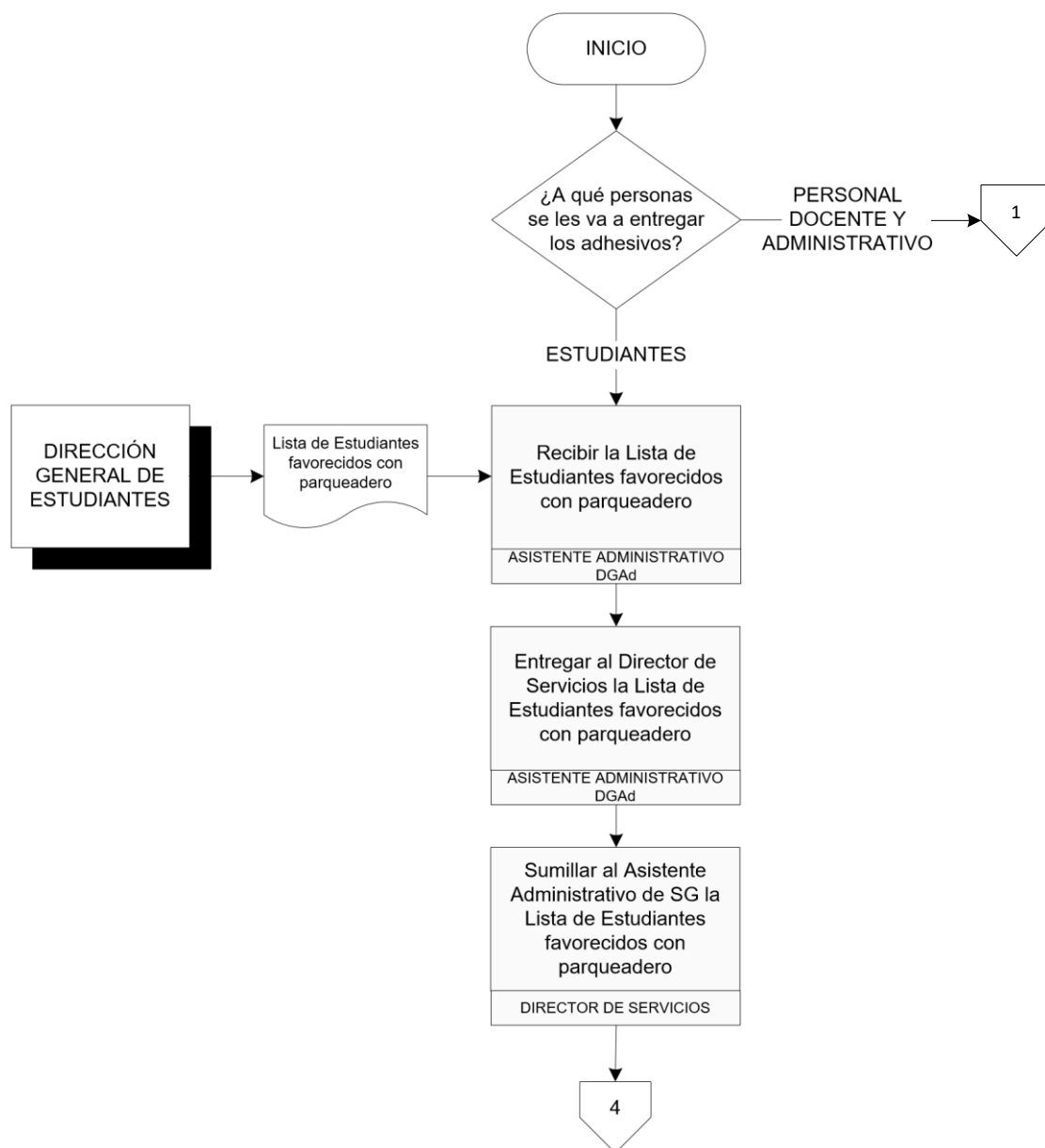
Se sugiere aumentar la muestra de la población utilizada para obtener un mejor análisis que incluyan usuarios que tengan plazas fijas para determinar el grado de aceptación del aplicativo.

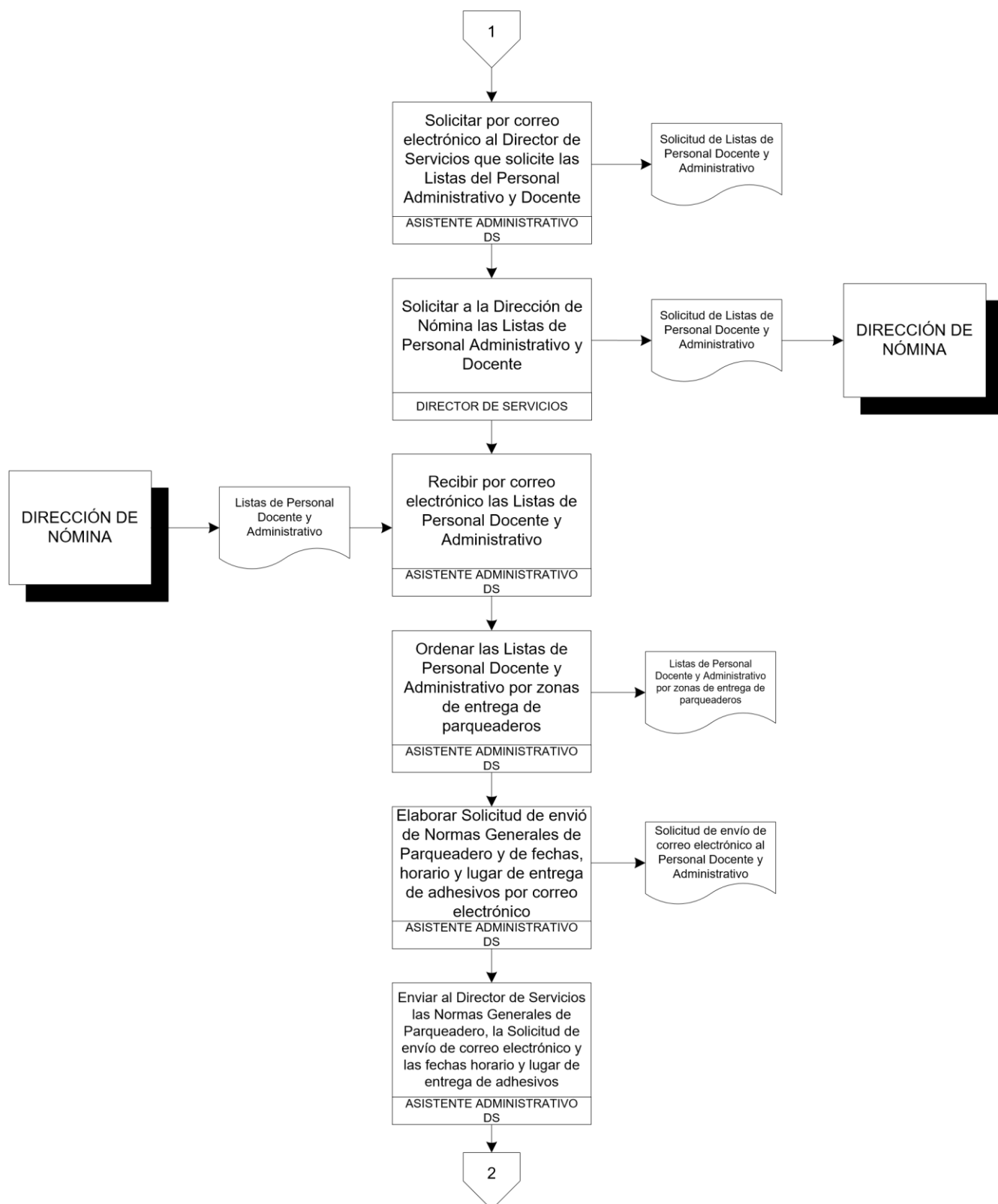
7.3 ANEXOS

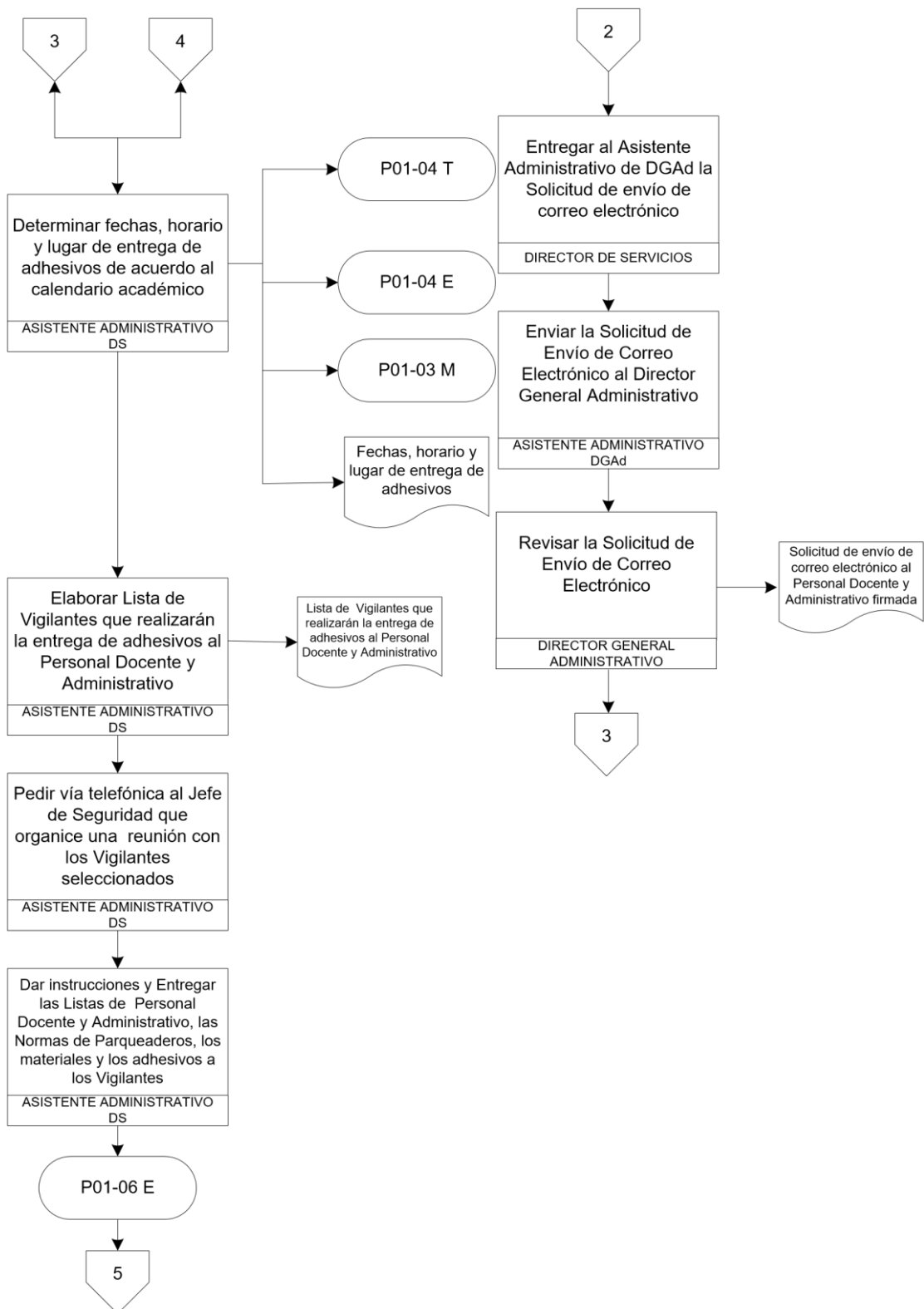
7.3.1 Anexo 1: Diagnóstico de procesos de la Dirección General Administrativa

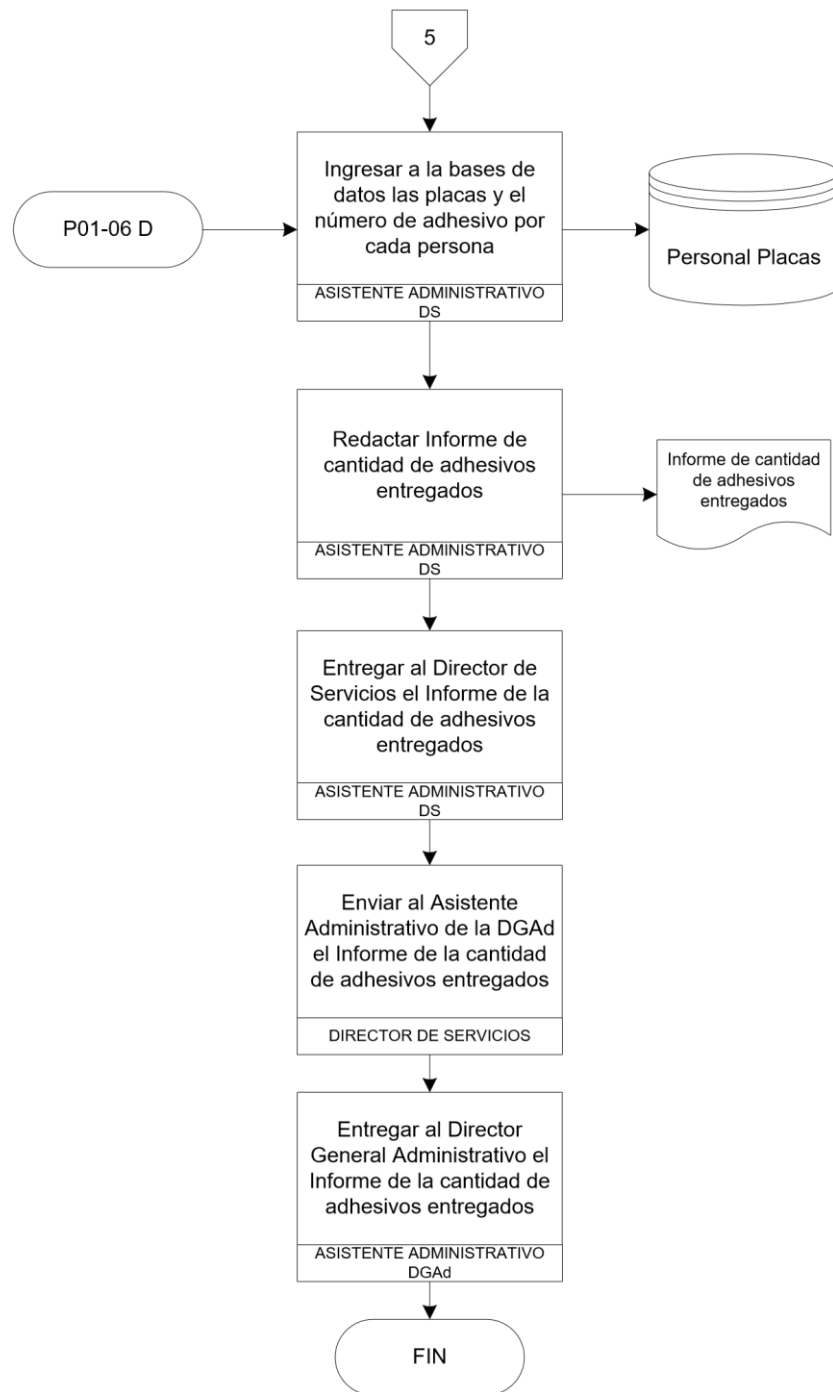
Proceso

ENTREGA DE ADHESIVOS DE PARQUEADERO









7.3.2 Anexo 2: Script de generación de la base de datos

-- phpMyAdmin SQL Dump

-- version 4.5.1

-- http://www.phpmyadmin.net

--

-- Servidor: 127.0.0.1

-- Tiempo de generación: 04-10-2016 a las 17:32:52

-- Versión del servidor: 10.1.10-MariaDB

-- Versión de PHP: 5.6.15

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";

SET time_zone = "+00:00";

/*!40101 SET

@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;

/*!40101 SET

@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;

```
/*!40101 SET
```

```
@OLD_COLLATION_CONNECTION=@ @COLLATION_CONNECTION */;
```

```
/*!40101 SET NAMES utf8mb4 */;
```

```
--
```

```
-- Base de datos: `bdd_parqueaderos`
```

```
--
```

```
-----
```

```
--
```

```
-- Estructura de tabla para la tabla `fechasdisponibles`
```

```
--
```

```
CREATE TABLE `fechasdisponibles` (
```

```
  `idFechaDisponible` int(11) NOT NULL,
```

```
  `fechaFechaDisponible` date NOT NULL,
```

```
  `horarioInicioFechaDisponible` time NOT NULL,
```

```
  `horarioFinFechaDisponible` time NOT NULL
```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
--
```

```
-- Volcado de datos para la tabla `fechasdisponibles`
```

```
--
```

```
INSERT INTO `fechasdisponibles` (`idFechaDisponible`, `fechaFechaDisponible`,  
`horarioInicioFechaDisponible`, `horarioFinFechaDisponible`) VALUES
```

```
(1, '2016-10-05', '07:00:00', '22:00:00');
```

```
-- -----
```

```
--
```

```
-- Estructura de tabla para la tabla `guardias`
```

```
--
```

```
CREATE TABLE `guardias` (
```

```
  `idGuardia` int(11) NOT NULL,
```

```
  `nombreGuardia` varchar(100) NOT NULL,
```



```
`apellidoGuardia` varchar(100) NOT NULL,  
  
`cedulaGuardia` varchar(10) NOT NULL,  
  
`idTipoUsuarios` int(11) NOT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Estructura de tabla para la tabla `horarios`

--

```
CREATE TABLE `horarios` (  
  
  `idHorario` int(11) NOT NULL,  
  
  `inicioHorario` time NOT NULL,  
  
  `finalHorario` time NOT NULL,  
  
  `descripcionHorario` varchar(500) NOT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Volcado de datos para la tabla `horarios`

--

```
INSERT INTO `horarios` (`idHorario`, `inicioHorario`, `finalHorario`,  
`descripcionHorario`) VALUES
```

```
(1, '07:00:00', '08:00:00', 'no aplica'),
```

```
(2, '08:00:00', '09:00:00', 'no aplica'),
```

```
(3, '09:00:00', '10:00:00', 'no aplica'),
```

```
(4, '10:00:00', '11:00:00', 'no aplica'),
```

```
(5, '11:00:00', '12:00:00', 'no aplica'),
```

```
(6, '12:00:00', '13:00:00', 'no aplica'),
```

```
(7, '13:00:00', '14:00:00', 'no aplica'),
```

```
(8, '14:00:00', '15:00:00', 'no aplica'),
```

```
(9, '15:00:00', '16:00:00', 'no aplica'),
```

```
(10, '16:00:00', '17:00:00', 'no aplica'),
```

```
(11, '17:00:00', '18:00:00', 'no aplica'),
```

```
(12, '18:00:00', '19:00:00', 'no aplica'),
```

```
(13, '19:00:00', '20:00:00', 'no aplica'),
```

```
(14, '20:00:00', '21:00:00', 'no aplica'),
```

(15, '21:00:00', '22:00:00', 'no aplica'),

(16, '22:00:00', '23:00:00', 'PM');

--

-- Estructura de tabla para la tabla `informes`

--

```
CREATE TABLE `informes` (  
  
  `idInforme` int(11) NOT NULL,  
  
  `tipoInforme` varchar(400) NOT NULL,  
  
  `fotoInforme` varchar(400) NOT NULL,  
  
  `descripcionInforme` varchar(500) NOT NULL,  
  
  `nombreUsuario` varchar(100) NOT NULL,  
  
  `idVehiculo` varchar(7) NOT NULL,  
  
  `fechaInforme` date NOT NULL,  
  
  `horaInforme` time NOT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Volcado de datos para la tabla `informes`

--

```
INSERT INTO `informes` (`idInforme`, `tipoInforme`, `fotoInforme`,
`descripcionInforme`, `nombreUsuario`, `idVehiculo`, `fechaInforme`,
`horaInforme`) VALUES
(1, 'Djdjdj', 'image.jpg', 'Sssss', 'gr', '1', '2016-10-04', '16:16:51');
```

--

-- Estructura de tabla para la tabla `logs`

--

```
CREATE TABLE `logs` (
  `idUsuario` int(11) NOT NULL,
  `fecha` date NOT NULL,
```

```
`hora` time NOT NULL,  
  
`descripcion` varchar(100) NOT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Estructura de tabla para la tabla `movimientos-vehiculares`

--

```
CREATE TABLE `movimientos-vehiculares` (  
  
  `idMovimiento` int(11) NOT NULL,  
  
  `vehiculoPlacas` varchar(7) NOT NULL,  
  
  `horaIngreso` time NOT NULL,  
  
  `horaSalida` time NOT NULL,  
  
  `fecha` date NOT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Estructura de tabla para la tabla `plazas`

--

```
CREATE TABLE `plazas` (  
  
  `idPlaza` int(11) NOT NULL,  
  
  `nombrePlaza` varchar(100) NOT NULL,  
  
  `idSector` int(11) NOT NULL  
  
  ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Volcado de datos para la tabla `plazas`

--

```
INSERT INTO `plazas` (`idPlaza`, `nombrePlaza`, `idSector`) VALUES  
  
(1, 'P1-ING', 1),  
  
(2, 'P2-ING', 1),  
  
(3, 'P3-ING', 1),
```

(4, 'P4-ING', 1),

(5, 'P5-ING', 1),

(6, 'P6-ING', 1),

(7, 'P7-ING', 1),

(8, 'P8-ING', 1),

(9, 'P9-ING', 1),

(11, 'P1-COL', 2),

(12, 'P2-COL', 2),

(13, 'P3-COL', 2),

(14, 'P4-COL', 2);

-- -----

--

-- Estructura Stand-in para la vista `plazaxreserva`

--

CREATE TABLE `plazaxreserva` (

`idPlaza` int(11)

, `nombrePlaza` varchar(100)

```
,`idSector` int(11)

,`idReserva` int(11)

,`fechaFechaDisponible` date

,`7-8` varchar(7)

,`8-9` varchar(7)

,`9-10` varchar(7)

,`10-11` varchar(7)

,`11-12` varchar(7)

,`12-13` varchar(7)

,`13-14` varchar(7)

,`14-15` varchar(7)

,`15-16` varchar(7)

,`16-17` varchar(7)

,`17-18` varchar(7)

,`18-19` varchar(7)

,`19-20` varchar(7)

,`20-21` varchar(7)

,`21-22` varchar(7)

);
```

--

-- Estructura de tabla para la tabla `reservas`

--

```
CREATE TABLE `reservas` (  
  
  `idReserva` int(11) NOT NULL,  
  
  `fechaFechaDisponible` date NOT NULL,  
  
  `idPlaza` int(11) NOT NULL,  
  
  `7-8` varchar(7) DEFAULT '0',  
  
  `8-9` varchar(7) NOT NULL DEFAULT '0',  
  
  `9-10` varchar(7) NOT NULL DEFAULT '0',  
  
  `10-11` varchar(7) NOT NULL DEFAULT '0',  
  
  `11-12` varchar(7) NOT NULL DEFAULT '0',  
  
  `12-13` varchar(7) NOT NULL DEFAULT '0',  
  
  `13-14` varchar(7) NOT NULL DEFAULT '0',  
  
  `14-15` varchar(7) NOT NULL DEFAULT '0',
```


(3, '2016-10-05', 3, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0'),
(4, '2016-10-05', 4, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0'),
(5, '2016-10-05', 5, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', 'PHN0801', 'PHN0801',
'PHN0801', '0'),
(6, '2016-10-05', 6, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0'),
(7, '2016-10-05', 7, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0'),
(8, '2016-10-05', 8, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0'),
(9, '2016-10-05', 9, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0'),
(10, '2016-10-05', 11, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0'),
(11, '2016-10-05', 12, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0'),
(12, '2016-10-05', 13, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0'),
(13, '2016-10-05', 14, '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0', '0');

-- -----

--

-- Estructura de tabla para la tabla `sectores`

--

```
CREATE TABLE `sectores` (  
  
    `idSector` int(11) NOT NULL,  
  
    `nombreSector` varchar(200) CHARACTER SET latin1 COLLATE  
latin1_spanish_ci NOT NULL,  
  
    `totalAbiertos` int(11) NOT NULL,  
  
    `totalCerrados` int(11) NOT NULL,  
  
    `totalMinusvalidos` int(11) NOT NULL,  
  
    `descripcionSector` varchar(300) NOT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Volcado de datos para la tabla `sectores`

--

```
INSERT INTO `sectores` (`idSector`, `nombreSector`, `totalAbiertos`,  
`totalCerrados`, `totalMinusvalidos`, `descripcionSector`) VALUES
```

```
(1, 'Parqueadero:Facultad de Ingeniería', 10, 2, 1, '0'),
```

```
(2, 'Parqueadero:Coliseo', 10, 2, 1, '0'),
```

```
(3, 'Parqueadero:Biblioteca', 10, 2, 1, '0'),
```

```
(4, 'Parqueadero:DISERLAB', 10, 2, 1, '0'),  
  
(5, 'Parqueadero:Edificio Administrativo', 10, 2, 1, '0'),  
  
(6, 'Parqueadero:Facultad de Ciencias Exactas', 10, 2, 1, '0'),  
  
(7, 'Parqueadero:Facultad de Arquitectura, Dise?o y Artes', 10, 2, 1, '0'),  
  
(8, 'Parqueadero:Gir3n', 10, 10, 10, ");
```

--

-- Estructura de tabla para la tabla `tipousuarios`

--

```
CREATE TABLE `tipousuarios` (  
  
  `idTipoUsuarios` int(11) NOT NULL,  
  
  `nombreTipoUsuarios` varchar(100) NOT NULL,  
  
  `descripcionTipoUsuarios` varchar(500) NOT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

```
-- Volcado de datos para la tabla `tipousuarios`
```

```
--
```

```
INSERT INTO `tipousuarios` (`idTipoUsuarios`, `nombreTipoUsuarios`,  
`descripcionTipoUsuarios`) VALUES
```

```
(1, 'Usuario', 'Actor capaz de generar una reserva puede ser estudiante o profesor.'),
```

```
(2, 'Administrador Master', 'Actor capaz de manejar casi todas las funcionalidades del  
sistema.'),
```

```
(3, 'Guardia de Control', 'Actor encargado de controlar los accesos y las salidas de cada  
uno de los vehículos.'),
```

```
(4, 'Guardia de Recorrido', 'Este actor recorrer los parqueaderos de la universidad .');  
  
-----
```

```
--
```

```
-- Estructura de tabla para la tabla `usuarios`
```

```
--
```

```
CREATE TABLE `usuarios` (
```

```

`idUsuario` int(10) NOT NULL,

`nombreUsuario` varchar(200) NOT NULL,

`apellidoUsuario` varchar(200) NOT NULL,

`idTipoUsuarios` int(10) NOT NULL,

`usuarioAccesoUsuario` varchar(100) NOT NULL,

`claveAccesoUsuario` varchar(200) NOT NULL,

`claseUsuario` varchar(200) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

--

-- Volcado de datos para la tabla `usuarios`

--

INSERT INTO `usuarios` (`idUsuario`, `nombreUsuario`, `apellidoUsuario`,
`idTipoUsuarios`, `usuarioAccesoUsuario`, `claveAccesoUsuario`, `claseUsuario`)
VALUES

(1, 'Angel Bernardo', 'Ruiz Andrade', 2, 'aruiz058', '57058827', ''),

(2, 'Nombre Guardia de recorrido', 'Apellido Guardia de recorrido', 4, 'gr', 'gr', ''),

(3, 'Nombre guardia de control', 'Apellido Guardia de control', 3, 'gc', 'gc', ''),

```

```
(7, 'Usuario', 'Usuario', 1, 'usr', 'usr', 'Usuario');
```

```
-----
```

```
--
```

```
-- Estructura Stand-in para la vista `usuariosxvehiculos`
```

```
--
```

```
CREATE TABLE `usuariosxvehiculos` (
```

```
  `idUserio` int(10)
```

```
, `nombreUsuario` varchar(200)
```

```
, `apellidoUsuario` varchar(200)
```

```
, `usuarioAccesoUsuario` varchar(100)
```

```
, `claveAccesoUsuario` varchar(200)
```

```
, `idTipoUsuarios` int(10)
```

```
, `idVehiculo` int(11)
```

```
, `vehiculoPlacas` varchar(7)
```

```
);
```

```
-----
```


--

-- Estructura de tabla para la tabla `vehiculos`

--

```
CREATE TABLE `vehiculos` (  
  
  `idVehiculo` int(11) NOT NULL,  
  
  `vehiculoPlacas` varchar(7) NOT NULL,  
  
  `idUserario` int(11) NOT NULL  
  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

--

-- Volcado de datos para la tabla `vehiculos`

--

```
INSERT INTO `vehiculos` (`idVehiculo`, `vehiculoPlacas`, `idUserario`) VALUES  
  
(1, 'PHN0801', 7);
```

--

-- Estructura para la vista `plazaxreserva`

--

DROP TABLE IF EXISTS `plazaxreserva`;

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL
SECURITY DEFINER VIEW `plazaxreserva` AS (select `p`.`idPlaza` AS
`idPlaza`,`p`.`nombrePlaza` AS `nombrePlaza`,`p`.`idSector` AS
`idSector`,`r`.`idReserva` AS `idReserva`,`r`.`fechaFechaDisponible` AS
`fechaFechaDisponible`,`r`.`7-8` AS `7-8`,`r`.`8-9` AS `8-9`,`r`.`9-10` AS `9-
10`,`r`.`10-11` AS `10-11`,`r`.`11-12` AS `11-12`,`r`.`12-13` AS `12-13`,`r`.`13-14`
AS `13-14`,`r`.`14-15` AS `14-15`,`r`.`15-16` AS `15-16`,`r`.`16-17` AS `16-
17`,`r`.`17-18` AS `17-18`,`r`.`18-19` AS `18-19`,`r`.`19-20` AS `19-20`,`r`.`20-21`
AS `20-21`,`r`.`21-22` AS `21-22` from (`plazas` `p` join `reservas` `r`
on((`p`.`idPlaza` = `r`.`idPlaza`))));
```

--

-- Estructura para la vista `usuariosxvehiculos`

--

DROP TABLE IF EXISTS `usuariosxvehiculos`;

```
CREATE ALGORITHM=UNDEFINED DEFINER=`root`@`localhost` SQL
SECURITY DEFINER VIEW `usuariosxvehiculos` AS select `u`.`idUsuario` AS
`idUsuario`,`u`.`nombreUsuario` AS `nombreUsuario`,`u`.`apellidoUsuario` AS
`apellidoUsuario`,`u`.`usuarioAccesoUsuario` AS
`usuarioAccesoUsuario`,`u`.`claveAccesoUsuario` AS
`claveAccesoUsuario`,`u`.`idTipoUsuarios` AS `idTipoUsuarios`,`v`.`idVehiculo` AS
`idVehiculo`,`v`.`vehiculoPlacas` AS `vehiculoPlacas` from (`usuarios` `u` join
`vehiculos` `v` on((`u`.`idUsuario` = `v`.`idUsuario`))) ;
```

--

-- Índices para tablas volcadas

--

--

-- Indices de la tabla `fechasdisponibles`

--

ALTER TABLE `fechasdisponibles`

```
ADD PRIMARY KEY (`idFechaDisponible`),
```

```
ADD UNIQUE KEY `fechaFechaDisponible` (`fechaFechaDisponible`);
```

```
--
```

```
-- Indices de la tabla `guardias`
```

```
--
```

```
ALTER TABLE `guardias`
```

```
ADD PRIMARY KEY (`idGuardia`);
```

```
--
```

```
-- Indices de la tabla `horarios`
```

```
--
```

```
ALTER TABLE `horarios`
```

```
ADD PRIMARY KEY (`idHorario`);
```

```
--
```

```
-- Indices de la tabla `informes`
```

```
--
```

```
ALTER TABLE `informes`
```

```
ADD PRIMARY KEY (`idInforme`);
```

```
--
```

```
-- Indices de la tabla `movimientos-vehiculares`
```

```
--
```

```
ALTER TABLE `movimientos-vehiculares`
```

```
ADD PRIMARY KEY (`idMovimiento`);
```

```
--
```

```
-- Indices de la tabla `plazas`
```

```
--
```

```
ALTER TABLE `plazas`
```

```
ADD PRIMARY KEY (`idPlaza`);
```

```
--
```

```
-- Indices de la tabla `reservas`
```

```
--
```

```
ALTER TABLE `reservas`
```

```
ADD PRIMARY KEY (`idReserva`);
```

--

-- Indices de la tabla `sectores`

--

ALTER TABLE `sectores`

ADD PRIMARY KEY (`idSector`);

--

-- Indices de la tabla `tipousuarios`

--

ALTER TABLE `tipousuarios`

ADD PRIMARY KEY (`idTipoUsuarios`);

--

-- Indices de la tabla `usuarios`

--

ALTER TABLE `usuarios`

ADD PRIMARY KEY (`idUsuario`);

--

-- Indices de la tabla `vehiculos`

--

ALTER TABLE `vehiculos`

ADD PRIMARY KEY (`idVehiculo`),

ADD UNIQUE KEY `vehiculoPlacas` (`vehiculoPlacas`),

ADD KEY `vehiculoPlacas_2` (`vehiculoPlacas`);

--

-- AUTO_INCREMENT de las tablas volcadas

--

--

-- AUTO_INCREMENT de la tabla `fechasdisponibles`

--

ALTER TABLE `fechasdisponibles`

MODIFY `idFechaDisponible` int(11) NOT NULL AUTO_INCREMENT,
AUTO_INCREMENT=2;

--

```
-- AUTO_INCREMENT de la tabla `guardias`
```

```
--
```

```
ALTER TABLE `guardias`
```

```
MODIFY `idGuardia` int(11) NOT NULL AUTO_INCREMENT;
```

```
--
```

```
-- AUTO_INCREMENT de la tabla `horarios`
```

```
--
```

```
ALTER TABLE `horarios`
```

```
MODIFY `idHorario` int(11) NOT NULL AUTO_INCREMENT,  
AUTO_INCREMENT=17;
```

```
--
```

```
-- AUTO_INCREMENT de la tabla `informes`
```

```
--
```

```
ALTER TABLE `informes`
```

```
MODIFY `idInforme` int(11) NOT NULL AUTO_INCREMENT,  
AUTO_INCREMENT=2;
```

```
--
```

```
-- AUTO_INCREMENT de la tabla `movimientos-vehiculares`
```

```
--
```



```
ALTER TABLE `movimientos-vehiculares`
```

```
MODIFY `idMovimiento` int(11) NOT NULL AUTO_INCREMENT;
```

```
--
```

```
-- AUTO_INCREMENT de la tabla `plazas`
```

```
--
```

```
ALTER TABLE `plazas`
```

```
MODIFY `idPlaza` int(11) NOT NULL AUTO_INCREMENT,  
AUTO_INCREMENT=15;
```

```
--
```

```
-- AUTO_INCREMENT de la tabla `reservas`
```

```
--
```

```
ALTER TABLE `reservas`
```

```
MODIFY `idReserva` int(11) NOT NULL AUTO_INCREMENT,  
AUTO_INCREMENT=14;
```

```
--
```

```
-- AUTO_INCREMENT de la tabla `tipousuarios`
```

```
--
```

```
ALTER TABLE `tipousuarios`
```

```
MODIFY `idTipoUsuarios` int(11) NOT NULL AUTO_INCREMENT,  
AUTO_INCREMENT=5;
```

```
--
```

```
-- AUTO_INCREMENT de la tabla `usuarios`
```

```
--
```

```
ALTER TABLE `usuarios`
```

```
MODIFY `idUsuario` int(10) NOT NULL AUTO_INCREMENT,  
AUTO_INCREMENT=8;
```

```
--
```

```
-- AUTO_INCREMENT de la tabla `vehiculos`
```

```
--
```

```
ALTER TABLE `vehiculos`
```

```
MODIFY `idVehiculo` int(11) NOT NULL AUTO_INCREMENT,  
AUTO_INCREMENT=2;
```

```
/*!40101 SET  
CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET  
CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
```

```
/*!40101 SET  
COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

7.3.3 Anexo 3: Encuesta

1. “¿Usted posee o ha poseído un parqueadero de la PUCE?”
 - a. Sí
 - b. No
2. “¿Qué sectores ha tenido asignados para su vehículo?”
 - a. Parqueadero Facultad de Ingeniería
 - b. Parqueadero Coliseo
 - c. Parqueadero Biblioteca
 - d. Parqueadero DISERLAB
 - e. Parqueadero Edificio Administrativo
 - f. Parqueadero Ciencias Exactas
 - g. Parqueadero Facultad de Arquitectura, Diseño y Artes
 - h. Parqueadero Girón
3. “¿Alguna vez se ha visto afectado por alguno de los siguientes problemas?
Puede seleccionar varias respuestas.”
 - a. Falta de plazas en los parqueaderos
 - b. Largas filas de espera
 - c. Falta de control por las infracciones relacionadas con los
parqueaderos
 - d. Falta de seguridad en el ingreso al parqueadero
 - e. Mala distribución de las plazas por sector de parqueaderos
4. “¿Estaría dispuesto a utilizar un aplicativo web para reservar un espacio en el
parqueadero?”
 - a. Sí

- b. No
 - c. Tal vez
5. “¿Con que frecuencia reservaría parqueadero en la universidad por día?”
- a. Dos o más veces al día
 - b. Una vez al día
 - c. No reservaría un parqueadero
6. “¿Con que frecuencia por semana reservaría parqueadero en la universidad?”
- a. Una vez por semana
 - b. De dos a tres veces por semana
 - c. Más de tres veces por semana
7. “¿Desearía encontrar datos sobre infracciones cometidas en el aplicativo de reservas?”
- a. Sí
 - b. No
8. “¿Con cuánto tiempo de anticipación reservaría una plaza en los parqueaderos?”
- a. Con una semana o más tiempo de anticipación
 - b. Con unos días de anticipación
 - c. Con unas horas de anticipación
 - d. Antes de ir a la universidad
 - e. Al llegar a la Universidad

7.3.4 Anexo 4: Manual técnico

El documento se encuentra adjunto como “Manual Técnico”.

7.3.5 Anexo 5: Manual Usuario

El documento se encuentra adjunto como “Manual de Usuario”.

8 BIBLIOGRAFÍA

- (2016). Obtenido de Definición.de: <http://definicion.de/html/>
- (2016). Obtenido de iBiblio:
<https://www.ibiblio.org/pub/linux/docs/LuCaS/Manuales-LuCAS/doc-curso-html/doc-curso-html/x38.html>
- (2016). Obtenido de <https://es.wikipedia.org/wiki/AJAX>
- (2016). Obtenido de SearchNetworking:
<http://searchnetworking.techtarget.com/definition/asynchronous>
- (2016). Obtenido de Apachefriends.org:
<https://www.apachefriends.org/es/index.html>
- (2016). Obtenido de Es.wikipedia.org: <https://es.wikipedia.org/wiki/XAMPP>
- (2016). Obtenido de Wpbeginner.com: <http://www.wpbeginner.com/glossary/apache/>
- (2016). Obtenido de Apache.org: <https://www.apache.org/>
- (2016). Obtenido de Chrome.google.com:
<https://chrome.google.com/webstore/detail/ripple-emulator-beta/geelfhphabnejhdalkjhgipohgpdnoc>
- (2016). Obtenido de SublimeText: <https://www.sublimetext.com/3>
- (2016). Obtenido de jQuery.com: <https://jquery.com/>
- (2016). Obtenido de Es.wikipedia.org: <https://es.wikipedia.org/wiki/JQuery>
- (2016). Obtenido de W3schools.com: <http://www.w3schools.com/css/>
- (2016). Obtenido de Es.wikipedia.org:
https://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada
- argouml.tigris.org*. (2016). Obtenido de Argouml.tigris.org:
<http://argouml.tigris.org/?mode=All&pageNum=2&itemsPerPage=25>
- Argouml-stats.tigris.org* . (2016). Obtenido de <http://argouml-stats.tigris.org/documentation/quickguide-0.32/ch01.html>
- Busch, N. (2016). Obtenido de Uwe.pst.ifi.lmu.de:
<http://uwe.pst.ifi.lmu.de/teachingTutorialPresentationSpanish.html>
- Es.wikipedia.org*. (2016). Obtenido de <https://es.wikipedia.org/wiki/MySQL>
- Librosweb.es*. (2016). Obtenido de
http://librosweb.es/libro/javascript/capitulo_1.html

MagicDraw - EcuRed. (2016). Obtenido de Ecured.cu:

<https://www.ecured.cu/MagicDraw>

MySQL :: MySQL Workbench. (2016). Obtenido de Mysql.com:

<https://www.mysql.com/products/workbench/>

MySQL / The Most Popular Open-Source Database / Oracle. (2016). Obtenido de

Oracle.com: <http://www.oracle.com/us/products/mysql/overview/index.html>

Nomagic.com. (2016). Obtenido de

<http://www.nomagic.com/products/magicdraw.html>

Otto, M. (2016). Obtenido de Getbootstrap.com: <http://getbootstrap.com/>

PHP 5 Tutorial. (2016). Obtenido de W3schools.com.

Php.net. (2016). Obtenido de <http://php.net/manual/es/features.php>

PHP: Hypertext Preprocessor. (2016). Obtenido de PHP: Hypertext Preprocessor:

<https://secure.php.net/>

SAP Sybase PowerDesigner / product information website. (s.f.). Obtenido de

Powerdesigner.de: <http://powerdesigner.de/en/>

Uazuay.edu.ec. (2016). Obtenido de

https://www.uazuay.edu.ec/estudios/sistemas/lenguaje_iii/MAnnualJavaScript/caracteristicas.htm

W3schools.com. (2016). Obtenido de http://www.w3schools.com/xml/ajax_intro.asp